情報工学入門選書 4

ディジタル回路

大阪大学教授／工学博士
田村進一 著

株式会社 昭晃堂

情報工学入門選書 4

ディジタル回路

大阪大学教授／工学博士
田村進一 著

株式会社 昭晃堂

まえがき

半導体文明の隆盛に伴って，ディジタル回路の知識は電気系学生だけでなく，広く一般の理工系学生，エンジニアに要求されるようになってきた．

本書は理工系大学学生やエンジニアのためのディジタル回路の入門書・教科書として企画されたものである．最近のディジタル回路はICやLSIを使用することを前提としているが，それらを使いこなすためにはそれらの論理的特性だけでなく電気的特性も把握・理解しておく必要がある．本書はディジタルICを使うために必要な基礎的技術・知識を体系的に修得することを第一義に考えて述べている．

1，2章では基礎的事項として半導体物性，トランジスタの特性，論理関数などについて述べているが，それらはIC，LSIなどを理解するために必要最小限のものに止めた．

3章ではTTL，ECL，nMOS，CMOSなどのICの基本的構成と使用法について述べた．

4～9章ではゲート，フリップフロップ，カウンタ，演算回路などの基本原理と構成法，使用法について述べる．ICの例や使用法は主として74シリーズTTL，LS-TTL，HS-CMOS，A-CMOSなど，最近実際によく使われる素子を中心に記した．

10章ではICメモリの原理，使用法について述べた．ただし，マイクロプロセッサとの接続については回路例をあげるに止めた．

11章ではアナログ信号を取り扱う際に必要となるA/D，D/A変換器やアナログICについて述べた．

付録にはクロック発生法，LED，フォトカプラ，RS-232Cインタフェースな

ど，本文を補足する事項について述べた．

さらに，実験や回路解析を行うときの便宜のため，IC名からその機能，ピン配置等を引けるよう索引をつけた．

いくつかの章末には若干の演習問題をつけた．解答は巻末にまとめて記してある．

昭和62年9月

著者記す

目　次

1 半導体とトランジスタ

2 2値論理

3 集積回路

4 汎用基本 IC

5 フリップフロップ

6 ゲート MSI

7 カウンタ構成法

8 カウンタ MSI

9 演 算 回 路

10 記 憶 素 子

11　アナログ回路

1　半導体とトランジスタ

ディジタル回路においてはICやLSIを用いて回路を組むが，本章においてはそれらの電気的特性を知る上で必要な半導体の物性やダイオード，トランジスタの基本的性質について学ぶ．

1.1　半導体の基本的性質

半導体とは金属など良導体と絶縁物の中間の性質を持つ．たとえば，良導体である銅の**比抵抗**は常温25°Cで6.9×10^{-5} $\Omega\cdot$cmであるのに対し，半導体のそれはおよそ10^{-1}～10^{12} $\Omega\cdot$cmである．よく使われる半導体には**シリコン**(比抵抗$2.7\times10^{5}\Omega\cdot$cm)やゲルマニウム(比抵抗47 $\Omega\cdot$cm)などがある．絶縁体の比抵抗は$10^{12}\Omega\cdot$cm以上である．なお，金属では温度が上がると熱による格子振動の増加により電気抵抗が増加するのに対し，半導体では熱エネルギーにより電気を運ぶ**キャリア**が伝導帯に押し上げられ抵抗が減少する(図1.1参照)．

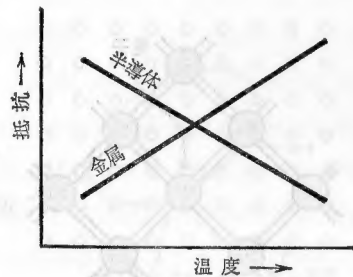
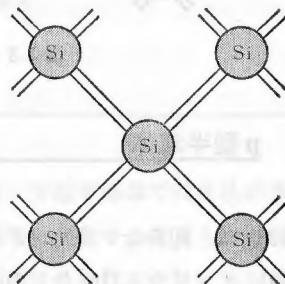



図 1.1　温度による抵抗の変化

図 1.2　シリコン結晶

シリコン(Si)の純粋な結晶では図1.2

に示すように，隣接するSi原子同士が電子を共有する共有結合をしている．

1.2　n型半導体

純粋なSiに5価の燐(P)，ひ素(As)，アンチモン(Sb)などを少量入れたものがn型半導体である．このとき，5価原子の最外殻電子(価電子)のうち4つは今までと同様に隣接するシリコン原子と共有結合をするが，残った1つの**価電子**は5価原子に束縛されてはいるが，その力は弱く，常温では容易に自由に動き回ることのできる**自由電子**となる(図1.3参照)．このように，主として電子が電荷を運ぶ役目(キャリア)をする半導体をn型半導体という．このとき，不純物として加えられた5価の原子は電子を与えるという意味で**ドナー**といわれる．

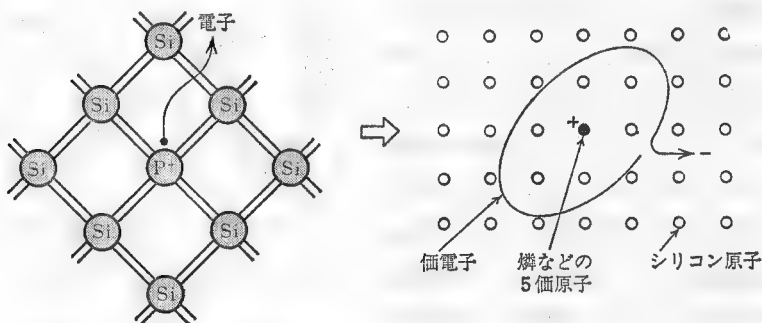



図 1.3　n型シリコン半導体

1.3　p型半導体

上とは逆に，純粋なシリコン半導体にホウ素(B)，アルミニウム(Al)，ガリウム(Ga)，インジウム(In)などの3価の原子を不純物として少量加えた場合を考えてみよう．このとき，3価の原子はやはりまわりのシリコン原子と共有結

合しようとするが，完全な共有結合をするには原子が1個不足する．そこで，これを埋め合わせるため隣りのシリコンの価電子を奪う．この結果，常温では3価原子のほとんどが他のシリコン原子の価電子を捕獲してマイナスイオンとなっている．そして，3価原子に捕獲された電子の抜けたあとは**正孔**となる．正孔はシリコン結晶内を共有結合から共有結合へと自由に動き回ることができる．それはあたかも正の電荷を持った粒子があるのと同じ働きをする(図1.4参照)．

このようなことから，3価の不純物を加えられた半導体は**多数キャリア**として正孔を持ち，p型半導体と呼ばれる．ここで，多数キャリアという言葉を使うのは，ほんのわずかではあるが自由電子が**少数キャリア**として存在するからである．正孔は自由電子と比較して少し動きにくい性質を持つ．このことが原因となり，p型半導体をベースにして素子をつくった場合，n型よりも動作速度が遅くなる．

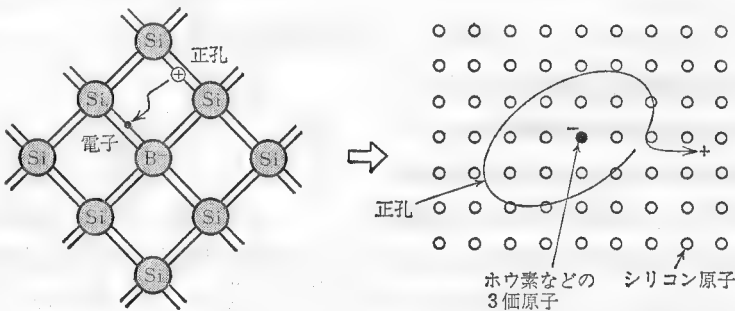



図 1.4 p型半導体

1.4 多数キャリア・少数キャリア

n型半導体では電子が多数キャリアであり，p型半導体では正孔が多数キャリアであることはすでに述べた．n型半導体では多数キャリアとして電子が存在するが，少数キャリアとしてごく少数ながら正孔も存在する．p型半導体においても同様である．この両方のキャリアが存在するため，バイポーラ・トラ

ンジスタにおける増幅作用が出てくるのである.

1.5　pn 接合

(a)　順方向電圧印加　　(b)　逆方向電圧印加

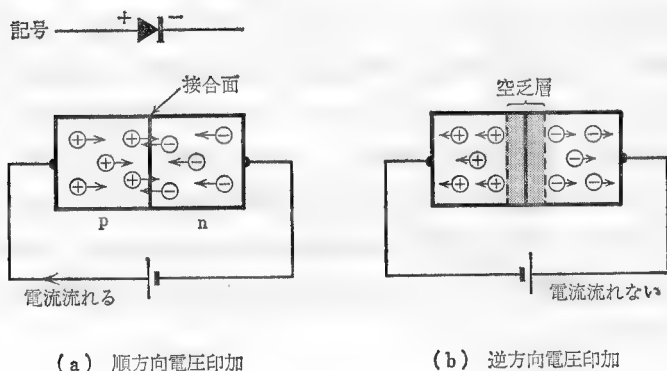
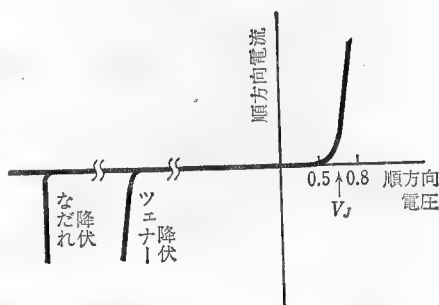



図 1.5　pn接合の整流作用

図 1.6　ダイオード特性

p 型半導体と n 型半導体が接合したダイオードは整流作用を持つ．すなわち，図1.5(a)のように順方向に電圧を加えると電流が流れるのに対し，同図(b)のように逆方向電圧を加えると電流はほとんど流れない．ダイオードの電圧·電流特性は図1.6のようになる．順方向に電流が流れ始めるときのpn接合電圧（導通電圧）V_J の大きさはおよそ 0.6～0.9 V と考えればよい．pn接合に逆方向電圧を加えたときには図1.5(b)に示されるようにpn接合部のキャリアの存在しない空乏層は厚くなり，電気的容量の性質を呈する．この容量の大きさは，逆方向印加電圧が大きいほど小さくなる．なお，順方向に電圧を印加したときでもその電圧が V_J 以下のときには，同様に容量性となる．

1.6　バイポーラ・トランジスタ

バイポーラとは両極，すなわちその動作に電子と正孔が関与することからこのように呼ばれる．通常，単にトランジスタといえば，このバイポーラ・トランジスタを指す．バイポーラ・トランジスタにはnpnタイプとpnpタイプがある．ここでは，よく使われるnpnタイプ・トランジスタについて述べる．

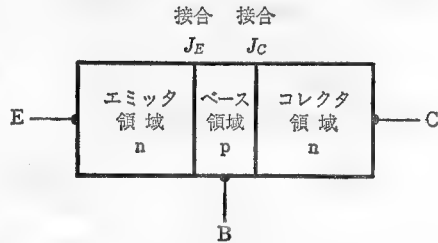



図 1.7　npnトランジスタ

npnトランジスタは図1.7に示すごとく，p型半導体でつくられた狭いベース領域をはさんで両側にn型半導体のエミッタ領域とコレクタ領域がある．図1.7において，左のn型領域はキャリア(電子)を放出するという意味で**エミッタ**(E)，中央のp型領域は**ベース**(B)，右のn型領域はキャリアを受け取るという意味で**コレクタ**(C)と名づけられている．図のように，いま**エミッタ-ベース接合**J_Eには順方向，**コレクタ-ベース接合**J_Cには逆方向電圧をかけるものとする．このとき，エミッタからベースに流れ込んだ電子のほとんど(たとえば99%)は勢い余ってそのままコレクタ領域に到達し，コレクタ電流となる．そして，ベースにはエミッタ電流のたとえば1%位が流れるのみである．

1.7　エミッタ接地

本節では，ディジタル回路で一番重要なトランジスタのエミッタ接地法について述べる．

いま，図1.8においてエミッタからベースへ流れ込んだ電子のうち，αがコレクタへ到達し，$(1-\alpha)$がベースに流れるとすると，

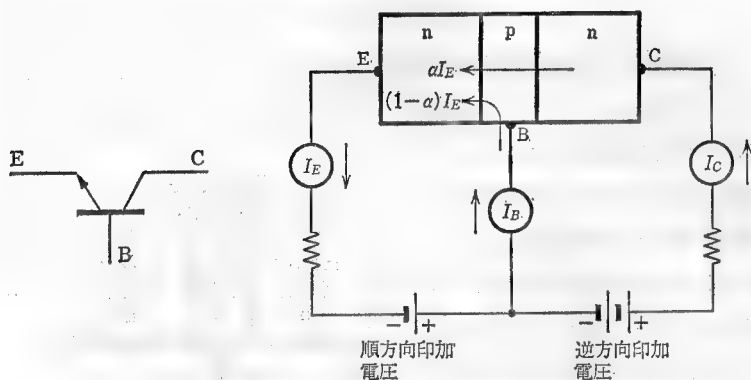
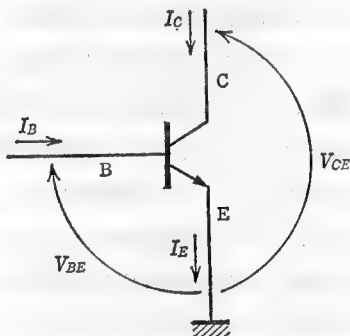



図 1.8　npn トランジスタの基本動作

図 1.9　エミッタ接地

$$I_C=\alpha I_E$$

$$I_B=(1-\alpha)I_E$$

となる．このαは，図1.8においてベースが接地されているとしたときの入力電流I_Eと出力電流I_Cの比であるから，**ベース接地直流電流増幅率**と呼ばれる．同様に図1.9において，ベース電流I_Bを入力，コレクタ電流I_Cを出力と考えると，**エミッタ接地直流電流増幅率**は，

$$\beta \triangleq \frac{I_C}{I_B}=\frac{\alpha}{1-\alpha}$$

となる．たとえば，もし，$\alpha=0.99$ならば，$\beta=99$となる．

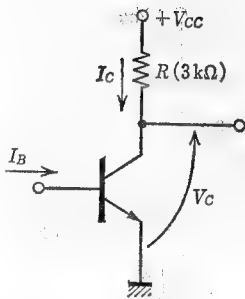
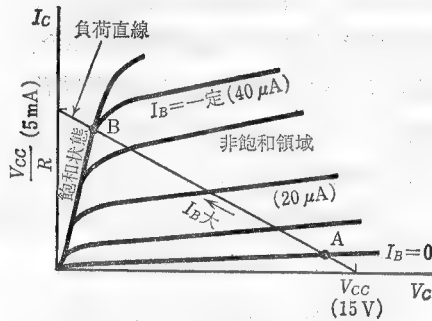
図1.10(a)のようなエミッタ接地でnpnトランジスタを使用する場合，同図(b)のような静特性が得られる．ベース電流$I_B=0$のときには動作点は点Aとなり，コレクタ電流I_Cはリーク電流しか流れない．これを**遮断状態**または**カットオフ状態**という．つぎに，ベース電流I_Bを順次増やしていくと動作点は負荷直線上を図のように移動していき，やがては点Bに達する．点Bに達するとI_Bを増やしても，もはやI_Cは増加しない．このような状態を**飽和状態**ま

(a)　回路図　　(b)　静特性

図 1.10　npn トランジスタのエミッタ接地静特性（(　)内は数値例を示す）

たは**オン状態**という．これらの中間の領域は非飽和領域である．多くのディジタル回路ではカットオフ状態と飽和状態の2つの状態を使って**スイッチング動作**をさせる．ただしこのような2状態を使用すると動作が安定で使いやすいのであるが，トランジスタを飽和させるため，**蓄積時間**(付録A.5参照)が増し，スイッチングに要する時間が増加する．飽和状態におけるトランジスタの各端子間の電圧はおおむね図1.11のようになっている．すなわち，$V_{BE}=0.7\text{V}$，$V_{CE}=0.2\text{V}$(コレクタ飽和電圧)と考えてよい．

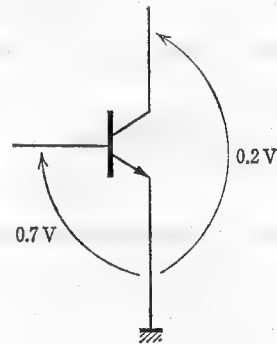



図 1.11　トランジスタの飽和状態における端子間電圧

1.8　エミッタホロワ

信号の振幅増幅よりも信号源のインピーダンス変換に用いられるトランジスタ回路に，図1.12に示すような**エミッタホロワ**回路がある．図1.12において，入力電圧 V_I が上がるとコレクタ電流，エミッタ電流が増加し，エミッタ電位 V_O も上昇する．したがって，V_O は V_I より pn 接合の順方向電圧である 0.7 V

程度低い電圧でV_Iに追随する．入出力電圧は同相(同じ方向に振れる)である．また，出力インピーダンスはエミッタ抵抗R以下となる．

一方，図1.12においては，

$$I_B=(1-\alpha)I_E$$

が成立するので，入力インピーダンスは，

$$R_I=\frac{V_I}{I_B}\fallingdotseq\frac{V_O}{I_B}=\frac{RI_E}{I_B}=\frac{R}{1-\alpha}$$

となり，大きな値となる．すなわち，信号の電圧振幅は変らないが，信号源としてのインピーダンスを$1/\beta$程度に低下できるのである．あるいは，別のいい方をすれば，小さな電流I_Bをそのβ倍程度のI_Eに増加できるのである．

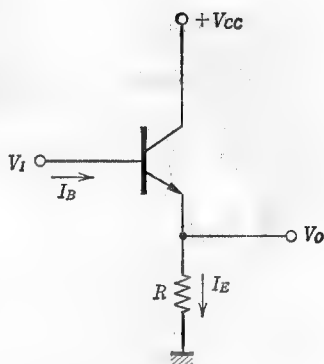



図 1.12　エミッタホロワ

1.9　MOS·FET

FETとは**電界効果型トランジスタ**(field effect transistor)のことで，多数キャリアのみを利用するため，バイポーラ・トランジスタに対して**ユニポーラ・トランジスタ**ともいわれる．FETには**接合型**，**MOS型**などがあるが，ICでは

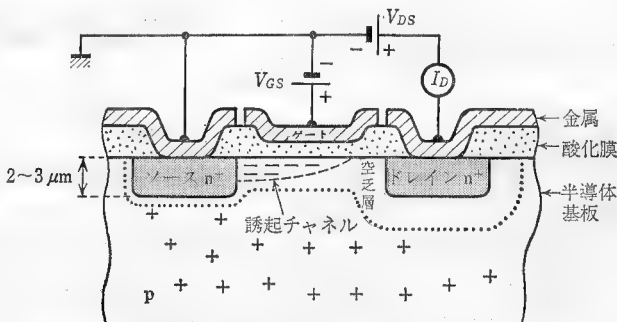



図 1.13　nMOS FETの構造(n^+とは不純物を多く入れたn型半導体)

主としてMOS型であるので，MOS·FETについて述べる.

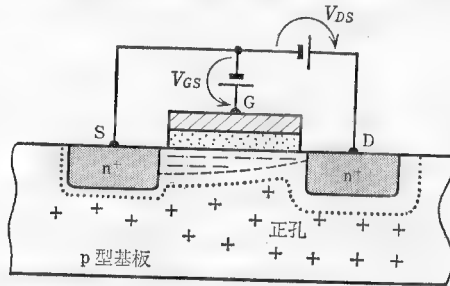
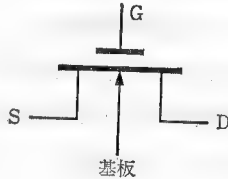
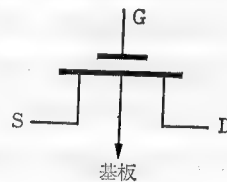
MOSとはMetal Oxide Semiconductor(**金属酸化物半導体**)の頭文字を取ったもので，**nチャネルMOS**·FETを例にとると図1.13のような構造を持つ.すなわち，p型半導体基板の上に**ソース**(S)と**ドレイン**(D)というn型領域を埋め込み，その上に絶縁物である薄い酸化膜(oxide)を形成させ，さらにその上に**ゲート**(G)と呼ばれる制御電極(metal)を置いた素子である.

(a) nチャネルMOS·FETの構造

(b) nMOS·FETの記号
(基板はしばしば省略される)

(c) pMOS·FETの記号

図 1.14 MOS·FET

図1.14に示すように，ゲートとソースの間にプラスの電圧V_{GS}を印加するとp型基板の多数キャリアである正孔は表面から追い出され，その代りに表面には少数キャリアである電子が現れてくる.この部分は伝導の形が逆転しているため**反転層**という.このとき，ドレインとソース間にドレインがプラスとなるように電圧をかけると，反転層に現れた電子がドレインに向って流れる.すなわち，ゲートに加える入力電圧によりソース-ドレイン間の基板表面にできる反転層を制御する.これにより，ソース-ドレイン間に流れる電流が変化す

る．ソースはその伝導にかかわるキャリアを放出し，ドレインはそれを吸収する．酸化膜は絶縁物であるからゲート電流はほとんど流れない．ただし，その厚さが薄く，静電気などにより絶縁破壊を起すことがあるのでその取扱いには注意を要する．なお，基板はソースに接続されることが多い．明示していない場合，ソースにつながれていると考えてよい．

図1.14（a）は V_{DS} が大きな飽和領域（次節参照）における状態を示している．V_{DS} を小さくする（非飽和領域）と空乏層は小さくなり，誘起チャネル（反転層）はドレイン（D）まで達するようになる．図1.14（a）は V_{DS} が小さな非飽和領域（次節参照）における状態を示している．V_{DS} を大きくする（飽和領域）と図1.13のように誘起チャネル（反転層）は途中（ピンチオフ点）で消えるが，空乏層を通してほぼ一定の電流が流れる．

nチャネルMOS（**nMOS**）とちょうど反対に構成された**pチャネルMOS**（pMOS）もつくることができる．この場合の伝導多数キャリアは正孔となる．nチャネルMOSでは伝導にかかわるキャリアは多数キャリアである電子である．正孔に比べて電子は**移動度**が大きいため，nチャネルMOSのほうがpチャネルMOSよりも動作速度の速い素子をつくることができる．歴史的にはpチャネルMOSのほうがつくりやすいため，素子の開発はnチャネルMOSに先んじて行われたが，性能を重視する集積回路ではnチャネルMOSのほうがよく使われるようになってきた．

1.10　nMOSの特性

MOS・FETのゲート電圧 V_{GS} を変えるとドレイン電流 I_D はどのように変化するであろうか．また，ドレイン電圧 V_{DS} との関係はどのようになっているであろうか．これらの様子を図1.15に示す．ゲート電圧を増すとドレイン電流 I_D も増すが，MOS・FETにはゲート電圧が零のときでもドレイン電流を流すことのできる**デプレション型**（D型）と流すことのできない**エンハンスメント型**（E型）とがある．なお，ドレイン電流が流れ始めるときのゲート電圧を**しき**

い値電圧 V_T という(図1.15(c)参照). したがって, D型MOSでは V_T は負, E型では V_T は正となる. これらはpMOS・FETについても同様である.

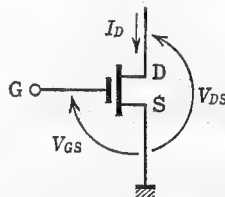
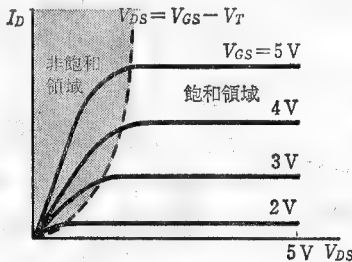
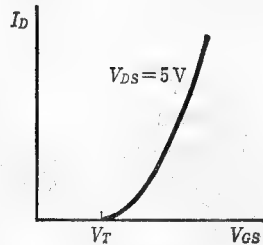
ゲート電圧がしきい値電圧 V_T より低いときにはドレイン電流が流れない($I_D=0$)のでトランジスタは遮断状態となる. 遮断領域ではチャネルが形成されていない. ゲート電圧がしきい値電圧 V_T より高くなるとソース-ドレイン間にはチャネルができ, 電流が流れはじめる. 電流が流れている状態には図1.15(b)に示すように**飽和領域**と**非飽和領域**がある. 飽和領域ではある一定値以上のドレイン電圧 V_{DS} に対してドレイン電流 I_D はほぼ一定となってしまう. なお, バイポーラ・トランジスタの V_{CE}, I_C, I_B をそれぞれMOS・FETの V_{DS}, I_D, V_{GS} に対応させてみた場合, 飽和と非飽和の対応がつかないが, それはバイポーラ・トランジスタでは, 増幅回路としての観点から, ベース電流を増やしてもコレクタ電流がそれ以上増加しない領域を飽和状態と称し, MOS・FETでは, 可変抵抗としての観点から, ドレイン電圧を増やしてもドレイン電流が

(a) 回路図

(b) n-MOS 静特性

(c) ゲート電圧とドレイン電流

図 1.15 nMOSトランジスタ静特性(数値はEタイプの例)

それ以上増加しない領域を飽和領域と称しているためである．観点が異なり変化させるものが対応しないのである．

演習問題

［1.1］　図1.16(a)，(b)において入力V_Iが各々10V，2Vのパルスのとき，出力V_Oを求めよ．なお，トランジスタの$\beta=100$とする．

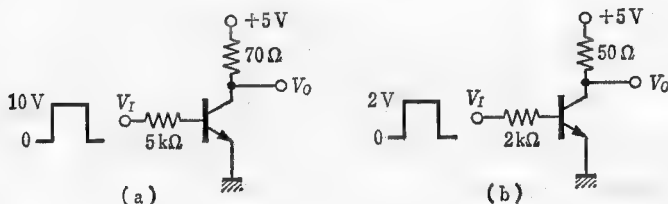



図 1.16

［1.2］　電流増幅率β_1，β_2を持つトランジスタを図1.17のように接続(**ダーリントン接続**)したとき，総合の電流増幅率βを求めよ．

［1.3］　図1.18(c)のような静特性を持つMOS·FETを(a)のように接続し，ゲートに5Vを与えたときの出力電圧V_{Oa}を求めよ．また(b)のように外部負荷がついたときはどうなるか．

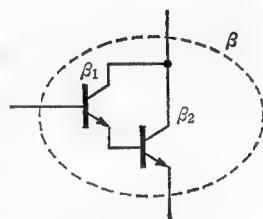



図 1.17　ダーリントン接続

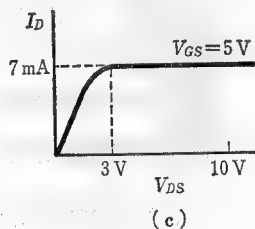
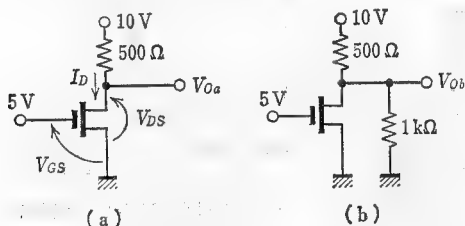



図 1.18

2　2値論理

ディジタル回路では通常2つの状態"0"と"1"を取り扱う．2つの論理値"真"(true)と"偽"(false)，またはそれに対応する"1"と"0"およびそれらの上に定義された演算からなる体系は**2値論理**(binary logic)と呼ばれる．2値論理は19世紀のイギリスの数学者George Boolにちなんで**ブール代数**(Boolean algebra)とも呼ばれる．2値論理の体系を用いると，ディジタル回路の特性をうまく表現できる．

2.1　基本論理演算

2値論理における基本論理演算には論理積，論理和，否定の3種がある．

(1)　論　理　積

2つの論理値AとBの論理積(logical product)は**AND**，**合接**(conjunction)とも呼ばれ，

$$f=A\cdot B$$

で表される．論理積記号"・"は省略されることもある．

表2.1　論理積の真理値表

A	B	f
0	0	0
0	1	0
1	0	0
1	1	1

論理積演算の真理値表を表2.1に示す．**真理値表**とは，論理値のいろいろな組合せに対して演算結果や関数がどのような値をとるかを記したものである．表2.1より，論理積では論理値1と1の組合せのときのみ値1をとり，他は0となることがわかる．

(2)　論　理　和

同じく論理和(logicalsum)は**OR**，離接(disjunction)とも呼ばれ，

$$f=A+B$$

で表される．この真理値表を表2.2に示す．すなわち，ともに0をとる論理値の和のみ値0となり，少なくとも一方が1をとる論理値の和は1となる．

表 2.2　論理和の真理値表

A	B	f
0	0	0
0	1	1
1	0	1
1	1	1

（3）否　　定

論理値を反転させる演算である．論理値Aの**否定**(negation)は**NOT**とも呼ばれ，

$$f=\bar{A}$$

と表現される．真理値表を表2.3に示す．

表 2.3　否定の真理値表

A	f
0	1
1	0

2.2　論理関数

論理関数(logical function)を表現するためには**論理変数**と**論理演算記号**を組み合わせた**論理式**(logical equation)が用いられる．たとえば，

$$f(X, Y)=X+Y \qquad (2.1)$$

$$f(X, Y, Z)=X\cdot Y+Z \qquad (2.2)$$

$$f(X, Y, Z)=(X+Y)\cdot Z \qquad (2.3)$$

などは論理式・論理関数である．ここでの演算は，式(2.2)のように論理積と論理和がある場合，論理積が先に行われ，つぎに論理和演算が行われる．また，式(2.3)のようにカッコがある場合はカッコ内演算が先に行われる．

表 2.4　双対性

論理積	論理和	否定
$0\cdot 0=0$	$1+1=1$	$\bar{0}=1$
$0\cdot 1=0$	$1+0=1$	$\bar{1}=0$
$1\cdot 0=0$	$0+1=1$	
$1\cdot 1=1$	$0+0=0$	

ある論理関係において，0と1，論理積と論理和を置き換えてできる論理関係をもとの論理関係の**双対**(dual)という．論理積と論理和の間には表2.4のような**双対性**(duality)がある．また，否定には**自己双対性**がある．

2.3　ベ　ン　図

ベン図(Venn diagram)とは図2.1に示すように，矩形領域を全世界とし，そのうちある論理関数が真となる領域を色をつけて示したものである．ベン図を用いると論理関数の性質が視覚的にわかりやすくなる．

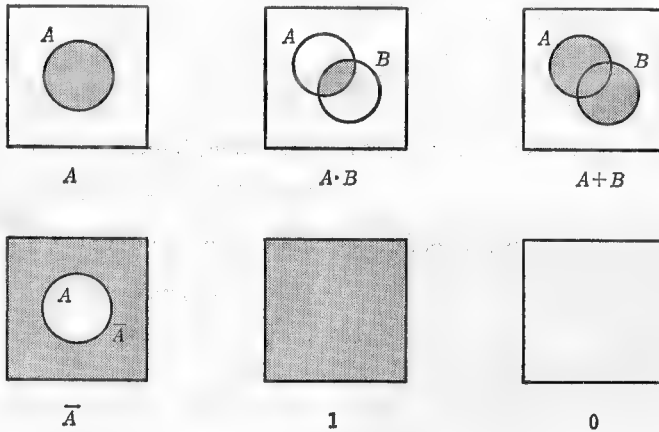



図 2.1　ベ　ン　図

2.4　定　　理

ブール代数におけるいくつかの定理を以下に列挙しておく．

定理 1　　$A\cdot A=A$

　　　　　$A+A=A$

定理 2　　$A\cdot 0=0$

　　　　　$A+1=1$

定理 3　　$\bar{\bar{A}}=A$

定理 4　　$A\cdot(A+B)=A$

$$A+A\cdot B=A$$

定理5　$A\cdot(\bar{A}\cdot B)=0$

$$A+(\bar{A}+B)=1$$

定理6　$A\cdot(\bar{A}+B)=A\cdot B$

$$A+\bar{A}\cdot B=A+B$$

定理7　$\bar{A}\cdot B=0$ かつ $\bar{A}+B=1$ ならば $A=B$ である．

定理8　$\left.\begin{array}{l}(A\cdot B)\cdot C=A\cdot(B\cdot C)\\(A+B)+C=A+(B+C)\end{array}\right\}$ 結合則

定理9　$\left.\begin{array}{l}\overline{A\cdot B}=\bar{A}+\bar{B}\\\overline{A+B}=\bar{A}\cdot\bar{B}\end{array}\right\}$ ド・モルガンの定理

定理10　$\left.\begin{array}{l}f(A,B,\cdots,K,\cdots,N)=K\cdot f(A,B,\cdots,1,\cdots,N)\\\qquad\qquad+\bar{K}\cdot f(A,B,\cdots,0,\cdots,N)\\f(A,B,\cdots,K,\cdots,N)=(K+f(A,B,\cdots,0,\cdots,N))\\\qquad\qquad\cdot(\bar{K}+f(A,B,\cdots,1,\cdots,N))\end{array}\right\}$ 展開定理

2.5　最小項と最大項

n変数の場合の変数値のとり得る値の組合せは2^n通りある．すべての変数またはその否定の論理積をとったものを**最小項**，また論理和をとったものを**最大項**という．いま，3変数X, Y, Zの場合について考えてみると，最小項，最大項は各8個あり，表2.5のようになる．最小項をベン図表示すると図2.2のようになる．また，最大項は全体から1個の最小項を取り除いたものになっていることがわかるであろう．たとえば，

表 2.5　最小項と最大項

変数値 X Y Z	最小項	最大項
0 0 0	$\bar{X}\cdot\bar{Y}\cdot\bar{Z}=m_0$	$\bar{X}+\bar{Y}+\bar{Z}=M_0$
0 0 1	$\bar{X}\cdot\bar{Y}\cdot Z=m_1$	$\bar{X}+\bar{Y}+Z=M_1$
0 1 0	$\bar{X}\cdot Y\cdot\bar{Z}=m_2$	$\bar{X}+Y+\bar{Z}=M_2$
0 1 1	$\bar{X}\cdot Y\cdot Z=m_3$	$\bar{X}+Y+Z=M_3$
1 0 0	$X\cdot\bar{Y}\cdot\bar{Z}=m_4$	$X+\bar{Y}+\bar{Z}=M_4$
1 0 1	$X\cdot\bar{Y}\cdot Z=m_5$	$X+\bar{Y}+Z=M_5$
1 1 0	$X\cdot Y\cdot\bar{Z}=m_6$	$X+Y+\bar{Z}=M_6$
1 1 1	$X\cdot Y\cdot Z=m_7$	$X+Y+Z=M_7$

$$M_2=\bar{X}+Y+\bar{Z}=\bar{m}_5 \tag{2.4}$$

であるから，M_2は全体から m_5を取り除いた領域である(図2.3参照)．一般には，ド・モルガンの定理を使って最小項と最大項にはつぎの関係があることがわかる．

$$\begin{cases} \bar{m}_k=M_{(2^n-1)-k} \\ \bar{M}_k=m_{(2^n-1)-k} \end{cases} \tag{2.5}$$

また，$k\neq l$のとき，

$$\begin{cases} m_k \cdot m_l=0 \\ M_k+M_l=1 \end{cases} \tag{2.6}$$

となる．なぜなら，異なる2つの最小項または最大項には少なくとも1つの変数の真と偽が含まれているからである．

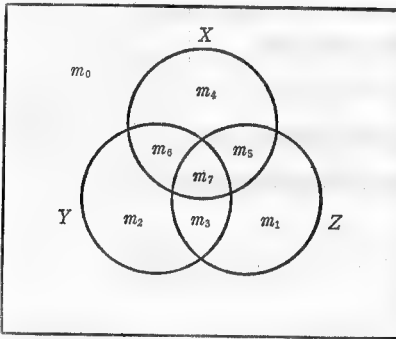



図 2.2 最小項のベン図

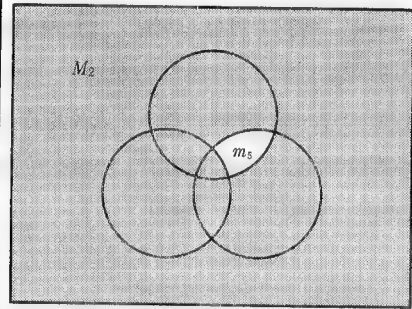



図 2.3 最大項$M_2=\bar{m}_5$

2.6 標 準 形

論理関数が論理変数の積の和の形で表されるものを**積和形**(sum of products form)といい，とくに，各積がすべての論理変数またはその否定からなる最小項である場合，その論理関数を**加法標準形**(disjunctive canonical form)または**標準積和形**(standard sum of products form)という．たとえば3変数X，Y, Z

の論理関数の加法標準形の例は,

$$f(X, Y, Z)=\bar{X}YZ+XYZ+\bar{X}\bar{Y}Z+XY\bar{Z} \qquad (2.7)$$

である.

加法標準形を求めるには，真理値表において関数値が1になるときの変数値が1のものはそのまま，0のものは否定をとって最小項を求め，それらの論理和をつくればよい.

【例】

表2.6に示される真理値をとる関数$f(X, Y, Z)$の加法標準形は,

$$f(X, Y, Z)=\bar{X}\bar{Y}Z+X\bar{Y}Z+XY\bar{Z}+XYZ$$

で与えられる.

表 2.6　真理値表

変数値 X	Y	Z	関数値 $f(X, Y, Z)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

一方，これと双対な形として，論理関数を論理変数の和の積の形で表す表し方がある．これを**和積形**(product of sums form)と呼ぶ．とくに各和が最大項になっているものを**乗法標準形**(conjunctive canonical form)または**標準和積形**(standard product of sums form) と呼ぶ．たとえば,

$$f(X, Y, Z)=(X+\bar{Y}+Z)\cdot(X+\bar{Y}+\bar{Z})\cdot(\bar{X}+Y+Z)$$

は3変数関数の乗法標準形である.

乗法標準形をつくるには，真理値表において関数値が0となる変数の値の組合せを取り出し，その中で変数値が0のところはそのまま，1のところはその否定をとり，それら全部の論理和(すなわち最大項)をつくり，さらにそのような論理和の積をとる．これが乗法標準形である.

2.7　カルノー図とベイチ図

上で得られた標準形は項の数が多く，そのまま論理回路で実現するには煩雑で冗長なことが多い．そこでこれを簡単化する手法について述べる．以下には

加法標準形の簡単化のみ記すが，双対性を利用すると乗法標準形の簡単化にも適用できる．

まず，ベイチ図，カルノー図のつくり方を説明する(図2.4参照)．**ベイチ図**(Veitch diagram)はベン図の領域をます目にかえたものである．**カルノー図**(Karnaugh diagram)はベイチ図の領域名の代りに論理値を各ます目につけたものである．このとき，隣り合うます目につけた各論理値は1個しか異ならないようにする．カルノー図では上端と下端，右端と左端は隣接していると考える．

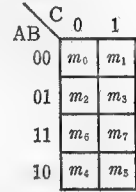
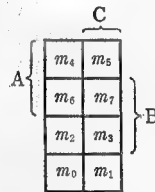
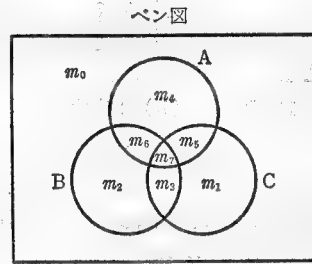



図 2.4 ベイチ図，カルノー図とベン図

このようにしてつくった1〜6変数の場合のカルノー図のます目を図2.5に示す．5変数以上になると立体的になるので，それを平面に展開した形で書く．つぎに，関数値が1の最小項に対応するます目に1を記入する．それ以外のます目は空白のまま，または0を記入しておく．

ここで，カルノー図を用いて論理関数を簡単化する方法について記す．まず，カルノー図上で1が隣接するます目は1×2，2×1，2×2，1×4，4×4など，$2^m \times 2^n (m, n=0, 1, 2, \cdots)$の単位でまとめる．その統合は重複してもよい．また，左右および上下の境界は反対側とつながっているとして考える．その例を図2.6に示す．このようにすると，その統合枠内には真・偽をともに含む変数があり，したがってその変数は消去することができる．たとえば，2個の1の組からは1個の変数を，4個の1の組からは2個の変数を，8個の1の組からは3個の変数を消去することができる．変数を消去したのち，1または1の組に対応する論理積の和をつくれば論理関数の簡単化ができる．図2.6の場合，もとの加法標準形，

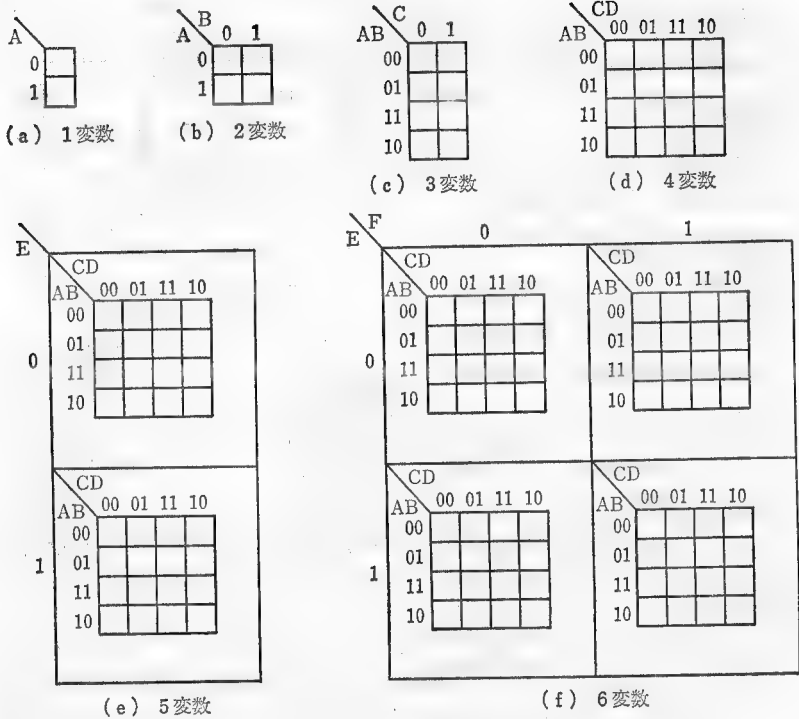



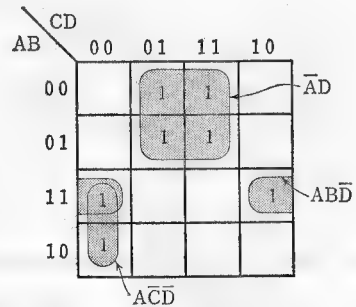
図 2.5　1～6 変数のカルノー図

$$f(A,B,C,D)=\bar{A}\bar{B}\bar{C}\bar{D}$$
$$+\bar{A}\bar{B}CD+\bar{A}B\bar{C}D$$
$$+\bar{A}BCD+AB\bar{C}\bar{D}$$
$$+ABC\bar{D}+A\bar{B}\bar{C}\bar{D} \quad (2.8)$$

に対し，簡単化した論理関数は，

$$f(A,B,C,D)=\bar{A}D$$
$$+AB\bar{D}+A\bar{C}\bar{D} \qquad (2.9)$$

となる．次節ではクワイン－マクラスキの方法による簡単化について述べる．

図 2.6　隣接する 1 の領域の統合

2.8　クワインーマクラスキの方法による論理式の簡単化

本節では図によらずに，論理式を加法標準形で表したとき最大項を順次まとめていくことにより簡単化を行うクワインーマクラスキの簡単化の方法について記す．この方法は変数が多くても扱え，かつ計算機で処理するのに適している．

表 2.7　クワインーマクラスキの方法における項のまとめあげ

	(a)	(b)
(1)	$\bar{X}\ \bar{Y}\ \bar{Z}$	$\bar{X}\ \bar{Y}$……A
(2)	$\bar{X}\ \bar{Y}\ Z$	$\bar{X}\ \bar{Z}$……B
	$\bar{X}\ Y\ \bar{Z}$	$\bar{Y}\ Z$……C
(3)	$X\ \bar{Y}\ Z$	$Y\ \bar{Z}$……D
	$X\ Y\ \bar{Z}$	$X\ Z$……E
(4)	$X\ Y\ Z$	$X\ Y$……F

例として，

$$f(X, Y, Z)=\bar{X}\,\bar{Y}\bar{Z}+\bar{X}\,Y\bar{Z}+\bar{X}YZ+\bar{X}\bar{Y}Z+XY\bar{Z}+XYZ \qquad (2.10)$$

を考える．表2.7(a)列のように加法標準形における項を否定形の個数の多いものから順に(1)〜(4)にグループ分けする．このとき，隣り合うグループから1個ずつ項を取り出し，それらの和をとることにより(b)列のように変数が減った項A〜Fをつくる．同じ項は何度使われてもよい．つぎに，(a)列を列項目，(b)列を行項目とする表2.8をつくり，(b)に含まれる(a)の個所に○印をつける．そして，すべての(a)が含まれるように(b)を選択する(◎印)．このようにして簡単化した，

$$f(X, Y, Z)=B+C+F=\bar{X}\bar{Z}+\bar{Y}Z+XY \qquad (2.11)$$

が得られる．一般に，簡単化の結果は唯一とは限らない．また，多変数の場合

表 2.8　クワインーマクラスキの方法における項の選択

(a) \ (b)	$\bar{X}\bar{Y}\bar{Z}$	$\bar{X}\bar{Y}Z$	$\bar{X}Y\bar{Z}$	$X\bar{Y}Z$	$XY\bar{Z}$	XYZ	選択された項
A	○	○					
B	○		○				◎
C		○		○			◎
D			○		○		
E				○		○	
F					○	○	◎

には簡単化した結果得られた項のいくつかをさらに簡単化するというように何段階かに分けて簡単化を行う.

演習問題

[2.1]　表2.9に示す真理値をとる3変数関数 $f(X, Y, Z)$ の加法標準形および乗法標準形を求めよ.

表 2.9　真理値表

変数値 X	Y	Z	関数値 $f(X, Y, Z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

[2.2]　カルノー図を用いて

$$f(A, B, C)=\overline{A}\overline{B}C+\overline{A}B\overline{C}+AB\overline{C}+ABC+A\overline{B}C$$

を簡単化せよ.

[2.3]　クワイン—マクラスキの方法により,

(1)　$f(X, Y, Z)=\overline{X}\overline{Y}Z+\overline{X}Y\overline{Z}+\overline{X}YZ+X\overline{Y}Z$

(2)　$f(X, Y, Z)=\overline{X}\overline{Y}\overline{Z}+\overline{X}\overline{Y}Z+X\overline{Y}\overline{Z}+X\overline{Y}Z$
$+XY\overline{Z}+XYZ$

を簡単化せよ.

3　集積回路

3.1　集積回路の種類

集積回路は1つの半導体基板上にすべての回路を構成することが多い．これを**モノリシック集積回路**(monolithic integrated circuit)と呼ぶ．単に集積回路とか**IC**といえばこれを指す．それに対して，セラミック等の基板上に能動回路チップやコンデンサ，印刷(10μmくらいの厚膜)もしくは蒸着(0.1μm以下の薄膜)された抵抗などの受動素子を混ぜて構成した集積回路は**ハイブリッド集積回路**(**混成集積回路**；hybrid IC)と呼ばれる．本書では主としてモノリシック集積回路について述べ，ハイブリッドICについてはふれない．

モノリシックICは通常シリコン基板上に構成される．その基本となるトランジスタにはバイポーラ・トランジスタとMOSトランジスタがある．バイポーラ・トランジスタを使用するバイポーラICには回路構成によりTTL，ECL，I^2L系がある．MOSトランジスタを使用するMOS ICにはpチャネルMOS(pMOS)，nチャネルMOS(nMOS)，相補形MOS(CMOS)がある．これ以外に，最近高速性にすぐれた新しい半導体材料として**ガリウムヒ素**(**GaAs**)を用いた素子が開発されてきているが，シリコンのように安定した絶縁膜が得られにくいため加工性に問題があり，普及には時間がかかると思われる．以上の分類に従ったICの種類分けを図3.1に示す．

IC評価の大きな要素は動作速度，消費電力，集積度である．**動作速度**は基本的なゲート素子(たとえばインバータ)1個の入出力間の**信号伝播遅延時間**

t_{pd}で表される．なお，遅延時間は振幅の50%の時間で計る．一般に動作速度は速いほど性能が高い．**消費電力**は基本ゲート素子1個の消費する電力Pで，消費電力は小さいほどよい．一般に動作速度を上げようとすれば消費電力は増

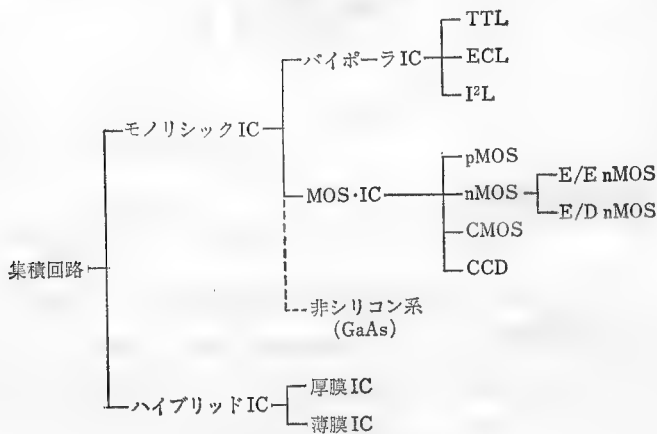



図 3.1 集積回路の種類

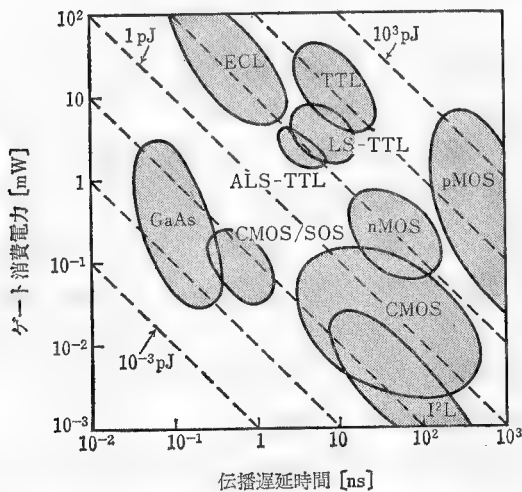



図 3.2 ICのゲート当りの消費電力と伝播遅延時間

加する傾向にある．そこで，両者を合わせて評価するため，**電力遅延時間積**（PT積またはPt_{pd}積）が使われる．これは小さいほどよい．図3.2に各種デバイスのゲート遅延時間，ゲート消費電力を示す．

TTL系に比べ，MOS系は速度や負荷駆動力の点では劣る場合も多いが，消費電力の点で優れている．ECLは消費電力が大きいが動作速度が速く，高速性が要求される場合に使われる．**CMOS/SOS**(CMOS silicon on sapphire)は通常のCMOSより消費電力は大きいが動作速度は1桁ほど速い．GaAsはCMOSに比べて消費電力が大きいが，高速性に非常にすぐれている．

集積度による分類

1つのチップ中に何個の抵抗やトランジスタなどの素子（またはゲート数）を含むかで集積回路は以下のような区分に分けられる．

チップ当りの集積度	区分
～100素子または～10ゲート	小規模集積回路**SSI** (small scale integration)
100～1000素子（10～100ゲート）	中規模集積回路**MSI** (medium scale integration)
1000～10^5素子（100～10^4ゲート）	大規模集積回路**LSI** (large scale integration)
10^5～10^7素子（64kbit DRAM, 16kbit SRAM以上）	超大規模集積回路**VLSI** (very large scale integration)
10^7～ 素子（16Mbit DRAM, 4Mbit SRAM以上）	超超大規模集積回路**ULSI** (ultra VLSI)

3.2 正論理・負論理

ディジタル回路では高い電圧（H）と低い電圧（L）の2つの電圧レベルで2つの状態を表す．このとき，高い電圧を論理"1"に，低い電圧を"0"に対応させる方法を**正論理**という．逆に，高い電圧を論理"0"に，低い電圧を"1"に対応させる方法を**負論理**という．すなわち，

	電圧レベル		論理
正論理	高(H)	⟷	“1”
(high active)	低(L)	⟷	“0”
負論理	高(H)	⟷	“0”
(low active)	低(L)	⟷	“1”

であり，物理的な性質であるH,Lを論理的な記号0と1に対応させるやり方が正論理／負論理である．

なお，**MIL記法**では，素子の入出力端子がlow active(Lが意味を持つ；負論理）であることを表すため○印を使う．回路の意図を明確にするため，原則として○印は○印と接続される．図3.3に例を示す．同図(b)において，ラインlのところは負論理であるが，論理的には同図(c)と全く同じものを表す．○印同士の対応がついている(b)の論理関数を求める場合，(c)のように○印を無視して求めればよい．

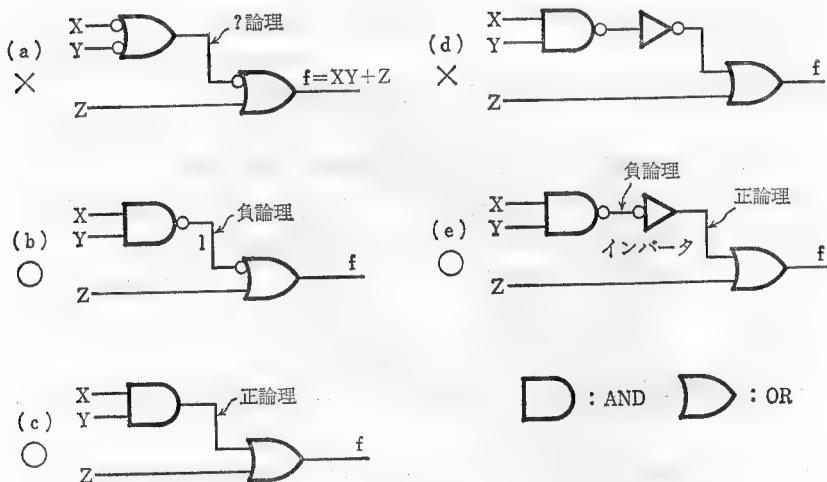



図 3.3 MIL記法(出力はすべて$f=XY+Z$.(a)と(b)，(c)は同じ論理関数を表すが，(b)，(c)のほうがより好ましい．同じく(e)は(d)より好ましい．)

3.3　ファンインとファンアウト

ファンイン(fan in)とは1個のICに接続できる入力数であり，**ファンアウト**(fan out)とは1個のICが駆動できる出力数である．これらは各々，扇(fan)のようにICに入っていく，または出ていく形からその名が由来している．ファンインはICのピン配置を見れば容易にわかる．ファンアウトは素子の流せる出力電流と次段素子の入力電流の比をHレベルとLレベルについて求め，それらの小さい方として直流的には決まる．しかしながら，とくにCMOSにおいては直流的なファンアウトは大きくても，その結果容量性負荷が増えると遅延時間が大きくなるので，高速動作が要求されるときにはこの面からも制限をうける．

3.4　ダイオード論理回路

図3.4のようにダイオードを接続すると，ダイオードの導通特性によりANDゲートやORゲートができる．ただし，ダイオードだけでは増幅作用を持たないため，多段につなげることはできない．

そこで，ダイオード論理回路の後にトランジスタをつけた**DTL**(diode tran-

(a)　ANDゲート　　(b)　ORゲート

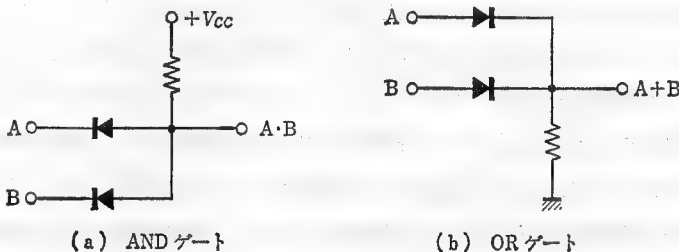



図 3.4　ダイオード論理回路

sistor logic)ICが現れたが，現在では次節にのべるTTLにとって代られた.

3.5　T　T　L

TTL(transistor transistor logic)はダイオードの働きをトランジスタのベース・エミッタ，あるいはベース・コレクタに置き換えたものである．図3.5に2入力NAND TTL ICの内部等価回路を示す．入力段トランジスタQ_1は**マルチエミッタ・トランジスタ**となっており，ダイオード論理回路とシングルエミッタ・トランジスタの組合せ(DTL)より高速動作が可能である．図3.5の入力部のダイオードは入力電圧が負になることを防ぐ保護ダイオードであり，論理操作と直接関係はない．AND機能はマルチエミッタが持っている．この回路の働きを説明する．

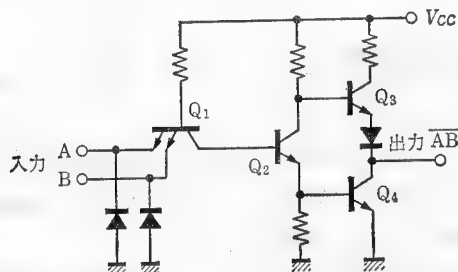



図 3.5　2入力NAND TTLゲート

入力がいずれもHレベル，たとえば3.7Vになった場合を考える．このとき，トランジスタQ_1にはエミッタとコレクタの原理的対称性に従って，図3.6のようにベース電流i_bが流れ，またエミッターコレクタ間にもわずかではあるが電流i_eが流れる．このi_b+i_eによりトランジスタQ_2が導通する．Q_2のONに従ってQ_4もONになる．一方，Q_3はベース電位が低いためOFFとなる．このときのおよその各部の電圧は図3.6に示されている．これはトランジスタ導通時のベース－エミッタ間電圧約0.7V，コレクターエミッタ間電圧約0.2Vから得られる．このときの出力はLレベル(0.2V)となる．

つぎに，入力の一方がLレベル，たとえば0.2Vになった場合を考える．このときQ_1は図3.7のように導通するのでQ_1のコレクタは低インピーダンス状態・低電位状態となる．そしてQ_2のベースに蓄積された過剰キャリアを引き出す．これによりQ_2は急速にOFFとなり高速動作が可能となる．Q_2がOFFとなる結果，Q_3はON，Q_4はOFFとなり出力端にはHレベルが出力される．入力端子がともにLレベルになっても動作は同じである．このことから，マルチエミッタ・トランジスタはAND特性を持つことがわかる．

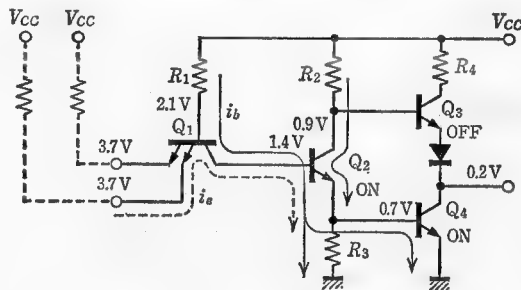



図 3.6 入力がともに高レベルの時

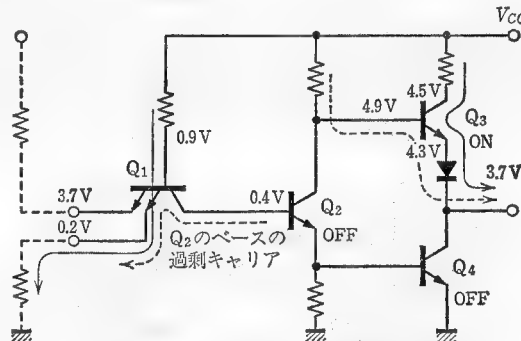



図 3.7 入力の少なくとも一方が低レベルの時

このICはHH入力に対して出力はLとなるのでANDの否定という意味で一般に**NAND**ゲートと呼ばれるが，これは入出力をともに正論理としたときの名称である．表3.1にその対応を示す．同表(b)のように，入力を正論理，出

力を負論理で考えるとこのゲートはANDの働きをする．また入力を負論理，出力を正論理とするとORとなる．

表 3.1　2入力NANDの入出力(A, Bは入力，*は負論理)

(a)物理的関係

A	B	出力
H	H	L
H	L	H
L	H	H
L	L	H

(b)論理的関係(入力が正論理のとき)

A	B	AND*	NAND
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	1

(c)論理的関係(入力が負論理のとき)

A*	B*	OR	NOR*
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

3.6　TTLの特性

TTLの出力段トランジスタの接続は図3.8(a)のようになっており，同図(b)と等価な働きする．TTLではファンアウトを大きくするため，標準的には同図(a)のような**トーテンポール形**出力回路を使用している．これにより，出力トランジスタQ_4がOFF(出力がHレベル)のときの出力インピーダンスがQ_3のONで低くなり，出力端子につながれた容量性負荷への充電を高速に行い，高速動作ができるようになっている．ダイオードDはQ_4がONのときpn接合の導通時電位差(およそ0.6V)を利用してQ_3のエミッタ電位を上げ，Q_3

(a)　トーテンポール出力　　　(b)　等価回路

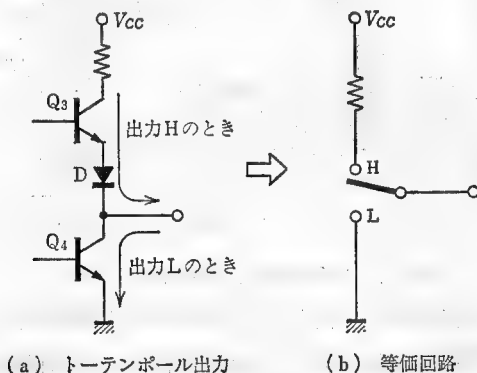
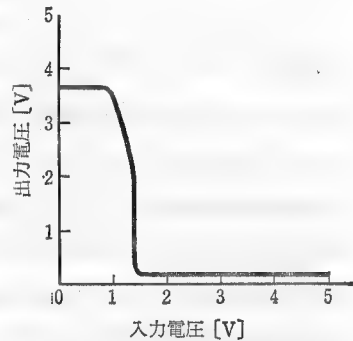



図 3.8　TTLの出力段と等価回路

図 3.9 TTLゲートの入出力特性

をOFFに保つようにするためのものである．なお，ダイオードを使用せず，その代りトランジスタ2個をエミッタホロワで縦続接続するTTL ICもある．トーテンポール接続のQ_3とQ_4は出力のL，Hに従ってどちらか一方のみ導通するようになっているが，出力がLからHに切り換るときに過渡的に数十mAのパルス電流が十数nsの時間流れる．これは，トランジスタQ_4は飽和しているのでキャリア蓄積時間だけOFFになるのが遅れるのに対し，トランジスタQ_3はエミッタホロワですぐに導通するからである．このパルス電流は電源ラインを通して他のラインのスパイク雑音となり妨害を与えるので，電源ラインに**バイパス・コンデンサ**を挿入してそれを吸収する必要がある．例として，ボードごとに10〜100μFタンタルコンデンサ1個または同容量電解コンデンサと0.1μFマイラコンデンサの組1つをつけ，さらにIC 5〜6個につき1つ0.1〜0.01μFのセラミックコンデンサをつける．

標準TTL (略号は N) ゲートの入出力等の特性を図3.9および3.10に示す．図3.10の表より，標準TTL同士を接続した場合のファンアウトは，

（i） 出力がHの場合

電源電圧	5V
入力電流 I_{IH}	40μA
入力電流 I_{IL}	1.6mA
出力電流 I_{OH}	0.4mA
出力電流 I_{OL}	16mA
平均消費電力 (duty 50%)	10mW/ゲート
平均ディレイ時間 t_{pd}	10ns/ゲート

図 3.10 標準TTLゲートの特性

$$\text{ファンアウト} = \frac{I_{OH}}{I_{IH}} = \frac{0.4\text{mA}}{40\mu\text{A}} = 10$$

（ii）　出力がLの場合

$$\text{ファンアウト} = \frac{I_{OL}}{I_{IL}} = \frac{16\text{mA}}{1.6\text{mA}} = 10$$

の小さい方(今はたまたま同じであるが)をとって，ファンアウトは10である．

一般にトランジスタを飽和させて使うと遅延時間が増加する原因となる．そこでこの飽和を浅くして遅延時間を小さくしようとするTTLが考案された．それは**Schottky TTL**(**S-TTL**と略)である．S-TTLでは図3.11に示すような**SBD**(Schottky barrier diode)でクランプされたトランジスタを使う．SBDは金属と半導体との接触によって生ずる電位障壁を利用したダイオードであり，

（i）　順方向電圧が0.3V位で低く，

（ii）　多数キャリアによる伝導を利用しているためキャリア蓄積現象がなくスイッチング動作が速い，

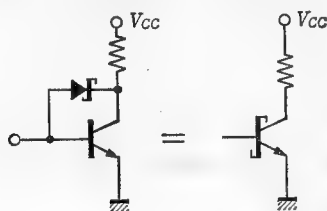
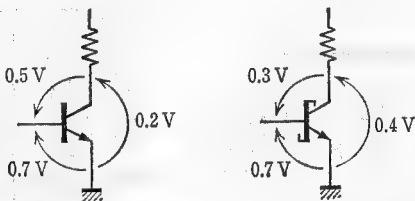
（iii）　製法上もコレクタの金属配線をベース領域に接触させるだけでよく特

（a）　SBDクランプトランジスタ回路

（b）　通常のトランジスタの飽和電圧　（c）　SBDクランプトランジスタの飽和電圧

図 3.11　SBDクランプトランジスタ

別な製造工程を必要としない．

などの特徴を持つ．このSBDを図3.11(a)のように接続したとき，ベース電圧が上がり，コレクタ電流が流れコレクタ電圧が下がって，コレクタ電圧とSBDの導通電圧(約0.3V)の和がベース-エミッタ間の導通電圧(約0.7V)より低くなるとベースへ流れるべき電流がSBDを通してコレクタ側へ分流し，ベース電流は過剰とならずトランジスタはちょうど飽和する程度に保たれる．S-TTLゲートの特性は伝播遅延時間3ns，消費電力19mW，直流的な特性は標準TTLと同じである．これらは表3.1にまとめられている．

S-TTLでは伝播遅延時間は改善されたが消費電力が増加している．そこで実用的にはS-TTLの消費電力を小さくした**LS-TTL**(low power S-TTL)がよく使われる．この消費電力は2mW，伝播遅延時間は9.5nsである．さらに，S-TTLやLS-TTLを微細加工技術でつくった**AS-TTL** (advanced S-TTL)や**ALS-TTL**(advanced LS-TTL)がつくられている．とくにALS-TTLは消費電力1mW，伝播遅延時間4nsであり，標準TTLより消費電力，遅延時間ともにすぐれたものとなっている．またAS-TTLは消費電力20mW，伝播遅延時間1.5nsである．

表 3.1　74シリーズTTL，CMOSの特性（*15 pF 負荷，**1 MHz）

	遅延時間〔ns〕	消費電力〔mW〕	出力電流〔mA〕		入力電流	
			I_{OH}	I_{OL}	I_{IH}	I_{IL}
標準 TTL	10	10	0.4	16	40 μA	1.6 mA
S-TTL	3	19	0.5	20	50 μA	2.0 mA
LS-TTL	9.5	2	0.4	8	20 μA	0.36 mA
HS-CMOS	9*	1.2**	4	4	1 μA	1 μA
A-CMOS	5	1.2**	24	24	1 μA	1 μA

3.7 E C L

ECL(emitter coupled logic；**エミッタ結合ロジック**)は別名**CML**(current mode logic；電流切換形論理回路)とも呼ばれ，**非飽和形論理回路**である．こ

の回路の特徴はトランジスタを飽和させないで使用するため，伝播遅延時間 t_{pd} は0.5〜2 nsと速いが，消費電力は30〜60 mW/ゲートとなっており，発熱が大きい．また，雑音余裕は0.5 VぐらいでTTLの1 Vに比べて小さく，雑音に対して弱い．

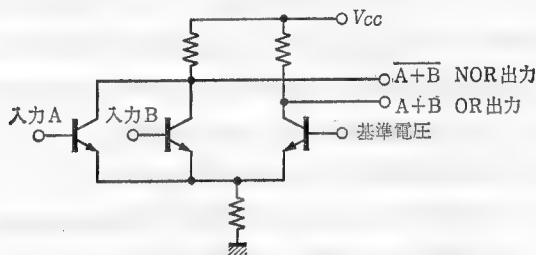



図 3.12　ECL(原理)

ECLは図3.12に示すように，左右のトランジスタのいずれかに常に電流を流しておき，左の入力信号によりこの電流を左右のトランジスタの間で切り換える．これによりスイッチング動作を行う．非飽和で動作させているため蓄積時間の影響がなく，高速で動作させることができる．基準電圧は入力電圧の高レベルと低レベルの中間に設定する．基準電圧は内部でつくることもできる．そのような回路例を図3.13に示す．また，V_{CC} をグランドレベルにし，V_{EE} に負電圧をかけたときの入出力特性を図3.14に示す．

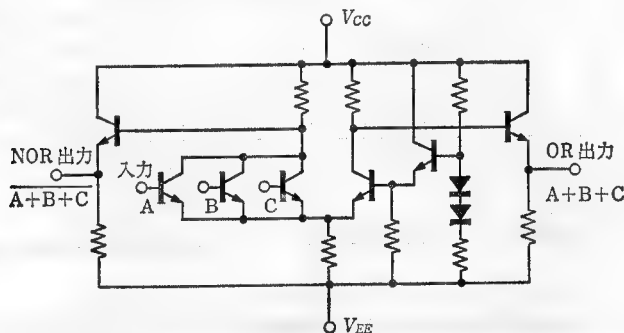



図 3.13　ECLゲート回路

ECLゲートは高速であり，単体として市販されているが，実際には雑音余裕が少ないなどの理由によりECLよりもAS-TTLなどを使う傾向にある．ECLは汎用よりもむしろ超大型計算機など高速性を要求される回路に専用LSIとして使用したほうがその特長を発揮しやすく，そのような使用法が多い．

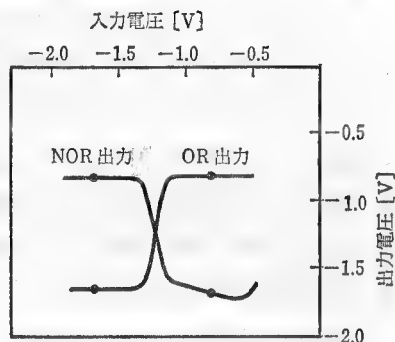



図 3.14　ECLゲートの入出力(V_{CC}=GND, V_{EE}=5.2V，•動作点，高レベル=−0.8V，低レベル=−1.6V)

3.8　nチャネルMOS IC

インバータ回路をnMOSトランジスタでつくることを考えよう．MOSトランジスタのON抵抗は数kΩであるので，ドライバMOSのON時の出力電圧(ドレイン電圧)を十分低く保つには負荷抵抗値を数十kΩにしなければならない(図3.15(a)参照)．このような高抵抗を拡散抵抗を用いてIC上に実現することは難しい．そこで，nMOSを負荷に使うことを考える．MOSトランジスタなら任意の抵抗値にコントロールするのは簡単である．ただし非直線性を持

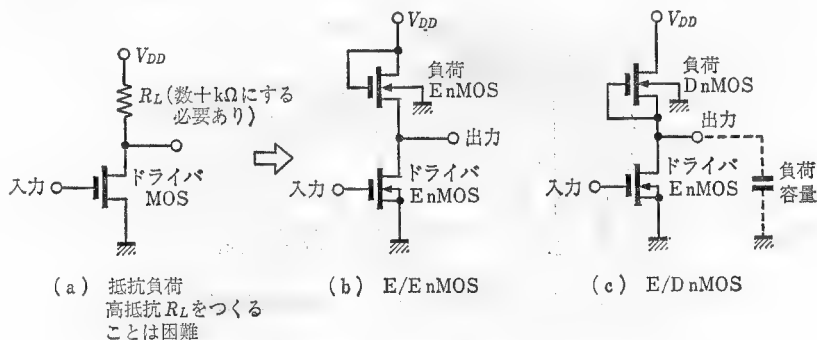



図 3.15　MOSインバータ

つ．その場合，ドライバMOSは入力が0のとき電流を流さないほうが望ましいのでエンハンスメント型(E nMOS)でつくる．図3.15(b)のように負荷MOSもE nMOSでつくるとドライバMOSと同じ工程でつくれるので簡単，安価となる．しかしながら負荷MOSのしきい値電圧V_TがプラスであるためドライバMOSのOFF状態におけるH出力は$V_{DD}-V_T$となり，信号振幅が小さくなるとともに負荷電流が少なくなって**ターンオフ時間**(出力がLからHになるときの伝播遅延時間)が長くなる．そこで，負荷MOSをD nMOSとした**E/D nMOS**がつくられた．

E/D nMOS(図3.15(c))ではドライバMOSがOFFのときのH出力は電源電圧V_{DD}まで上がり，信号振幅が大きく，かつターンオフ時間も速くなる．負荷D nMOSのゲートはソースと結ばれており，ゲート-ソース間電圧V_{GS}は零であり，定電流特性(図3.16)を持つ．すなわち，浮遊容量等の負荷容量を十分高い電圧まで一定の電流で短時間に充電することができ，E/E形よりターンオフ時間が短くなっている．一方，**ターンオン時間**はドライバ・トランジスタの特性で決まるためE/E，E/D形ともに同じ位の速さである．したがって，総合的にはE/D形のほうがE/E形よりすぐれており，よく使われる．

E/Dタイプ nMOSは高速動作が可能で，かつ大規模集積が可能なため，マイクロプロセッサや大規模メモリはほとんどこのタイプである．

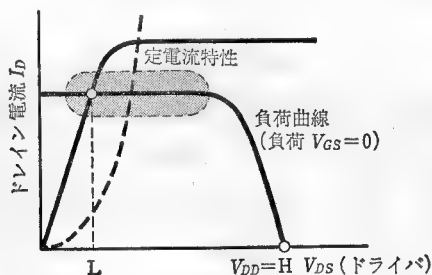



図 3.16 E/D nMOSインバータの負荷曲線と動作点

3.9 C M O S

インバータにおいてドライバnMOSトランジスタの負荷として図3.17(a)に示すように逆のpMOSトランジスタを使ったものが**CMOS**(complementary MOS；**相補形MOS**)である．このようにすると図3.17(b)に示すようにpMOSまたはnMOSのいずれかがONのとき他方はOFF状態となる．OFF状態のとき電流はほとんど流れないから，回路自体は直流状態では電力をほとんど消費しない．しかし入力電圧が変化して出力が反転する過渡的な状態ではpMOS, nMOSともにONとなり，一時的に電流が流れてしまう．また負荷の容量成分に対しても充放電がp, n両トランジスタによって行われるので負荷駆動能力は大きい．消費電力はこれらの理由によりスイッチング回数にほぼ比例する．以下にCMOSの特長，問題点を列挙する．

(1) **CMOSの特長**

(i) 電源の動作範囲が広く3〜16Vであり，1.5V位でも動作可能とすることができる．TTLの場合は5V±10%で非常に狭い．

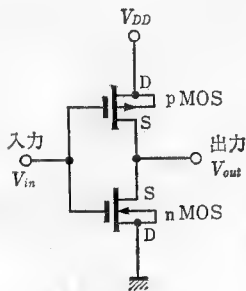
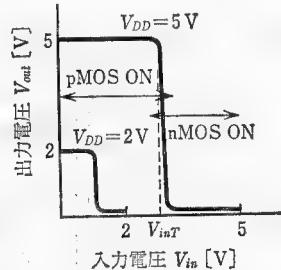
(ii) pMOSとnMOSの一方がON状態のとき他方は完全なOFF状態となっているから出力電圧の振幅は0〜V_{DD}と大きい．

(a) 回 路　　(b) 入出力電圧特性

図 3.17 CMOSインバータ

(iii) 入力電圧を上げていった場合，出力が反転する点(しきい値電圧 V_{inT})がほぼ電源電圧 V_{DD} の半分であり，その傾きは急峻である．したがって，雑音余裕度が大きい．

(iv) 消費電力がたとえば10μW/ゲートと非常に小さい．

(v) 入力制御のための電流をほとんど流す必要がないので出力のファンアウトを多くとることができる．

(vi) バイポーラと組み合わせることにより，同一基板上にアナログ回路とディジタル回路を共存させることが可能となる．

(2) **問題点・留意点**

(i) MOS-ICは半導体基板の表面方向にそって形成されるため，ドレイン-ソース間のON抵抗を小さくすることが難かしく，縦形のバイポーラと比較してスイッチング特性が悪い．

(ii) pMOS，nMOS異なるものを組み合わせているため，たとえばn形基板中にp形領域(pウェル)を形成し，その中にpMOSトランジスタをつくらねばならない．したがって素子間隔がつめられず，集積度を上げられない．

(iii) **過渡電流**：静的にはリーク以外に電力を消費せず電力消費は非常に少ないが，図3.18のように入力パルスが鈍っていると出力反転時に一時的にp, n両MOSがONとなり，過渡電流が流れ，かなり電力を消費する．また，図3.19に示すように次段CMOSのゲート容量も含め，浮遊容量などの容量性成分が多いと出力の立ち上りでpMOSから負荷容量へ

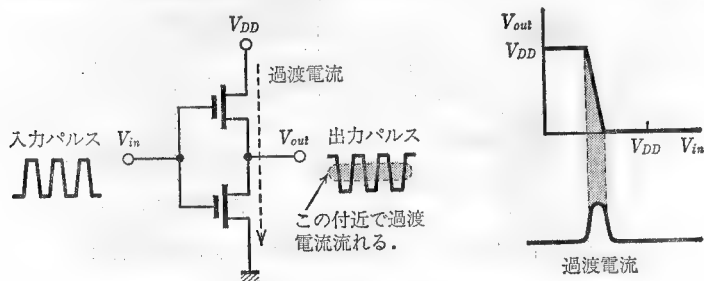



図 3.18 CMOSの出力反転時過渡電流

（a） 充放電電流　（b） 周波数特性

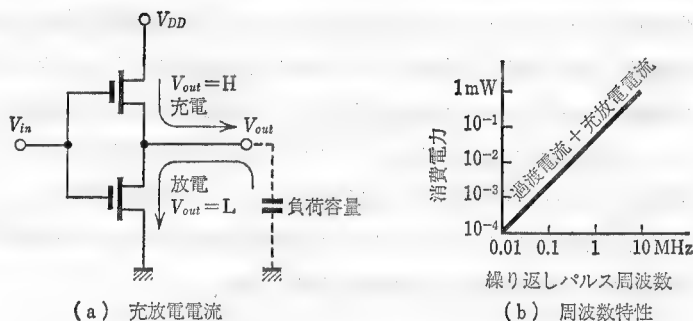
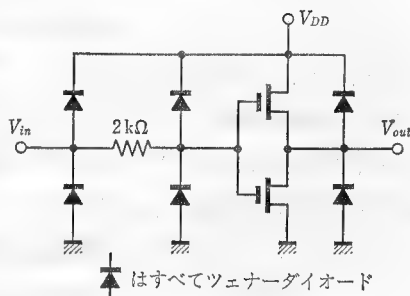



図 3.19 CMOSの消費電力

充電電流が流れ，立ち下りで放電電流が負荷容量からnMOSへ流入する．このため，入力パルスの繰り返し周波数が増加すると消費電力も増加する．

図 3.20 CMOSの絶縁破壊保護回路

（iv） **絶縁破壊**：ゲートのSiO_2絶縁膜は絶縁性が非常によいため，入力インピーダンスが高く，静電気（容易に数万V）による電荷がゲートに加わると絶縁破壊を起す．MOS酸化膜の絶縁破壊電圧はせいぜい100V程度である．一度絶縁破壊を起すと回復不可能なため，入力端子と出力端子には**ツェナーダイオード**（降伏電圧20V程度）と抵抗をつけて保護する（図3.20参照）．

（v） **ラッチアップ**：CMOSにおいては，入力パルスに重なった雑音電圧やオーバシュート，アンダシュートにより，V_{DD}より＋0.3V以上，またはGND電位より－0.3V以下になると，ICのV_{DD}とGND間に過渡電流が流れ，入力を取り除いても切れず，V_{DD}電源を一度切らないとも

とに戻らなくなる．これをラッチアップといい，CMOSの構造上，寄生のバイポーラ・トランジスタが形成され，その正帰還結合による現象でスイッチング素子であるサイリスタと同じ原理で生じる．ラッチアップが起こると素子をいためたり壊したりすることもある．構造的にラッチアップが起こりにくいよう工夫されてはいるが，完全に防止されているわけではないので異常電圧が加わらないよう**保護抵抗**(数kΩ)，保護ダイオードを入出力回路へ入れたり**電流制限抵抗**(数百Ω)を電源ラインへ入れるなどの保護対策を必要に応じて行う．

3.10 MOS論理回路

本節ではMOS ICのうち，よく使われるE/D nMOSとCMOSによる論理回路構成法について比較してみよう．

(1) NAND回路

図3.21(a)に示されるようにnMOSではE形nMOS 1と2が直列に接続されており，入力*A*, *B*がともにHレベルのときのみONとなり出力はLレベルとなる．負荷のD nMOSは負荷抵抗の役目を果しているだけであるから1個でよい．一方，CMOSの場合，負荷を入力状態により切り換える必要がある．すなわち，入力*A*, *B*のうち少なくとも一方がLレベルのとき，負荷のpMOSをON(出力はHレベル)にしなければならない．したがって，図のように負荷pMOSが並列に入っている．すなわち，CMOSでは入力1個に対してドライバMOSトランジスタとそれと対の負荷MOSトランジスタ各1個がいることになる．一方，nMOSでは負荷トランジスタは1個だけでよいから，入力数が多いときnMOSのほうがCMOSより素子数の点で有利となる．

(2) NOR回路

図3.21(b)に示されるように，nMOS，CMOSともにドライバ・ドランジスタは並列に並べるが，NANDの場合と同じくnMOSでは負荷トランジスタ1個ですむのに対し，CMOSでは入力の数と同じだけ負荷トランジスタが要る．

(3)　CMOSトランスミッション・ゲート

CMOSではpMOSとnMOSを並列に並べることにより，双方向性を持ったゲートができる．これを**トランスミッション・ゲート**と呼ぶ．図3.22において，ϕが高レベルのときp, n両トランジスタともにONとなり，数百Ω程度の

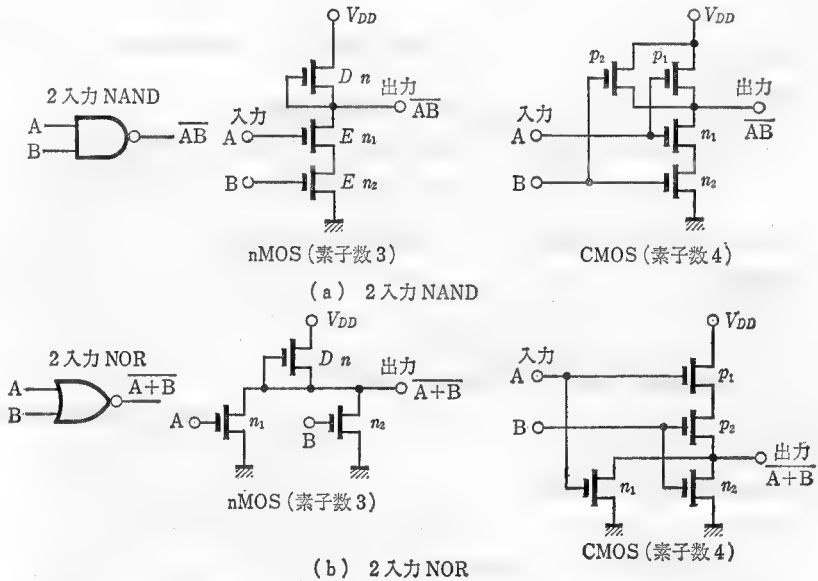



図 3.21　nMOSとCMOS論理回路

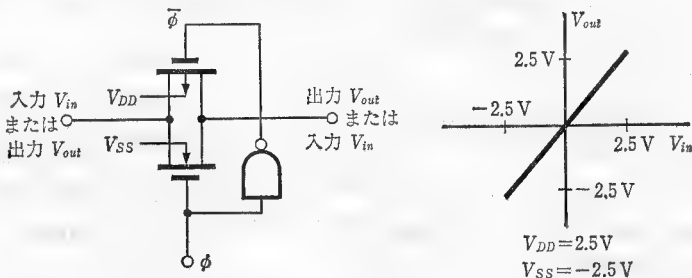



図 3.22　CMOSトランスミッション・ゲート

低抵抗で左右両端子間がつながる．ϕが低レベルになると両トランジスタともにOFFとなり，両端子間は切り離される．

演習問題

［3.1］ 74シリーズTTLおよびHS-CMOSの標準的な入出力電流は表3.2に示すような値である．

表3.2 74シリーズICの入出力電流

	入力電流		出力電流	
	I_{IL}	I_{IH}	I_{OL}	I_{OH}
標準 TTL	1.6mA	40μA	16mA	0.4mA
LS-TTL	0.4mA	20μA	8mA	0.4mA
HS-CMOS	−	−	4mA	4mA

（1） LS-TTLで標準TTLを駆動する場合

（2） HS-CMOSで標準TTLを駆動する場合

のファンアウトを求めよ．ただし，インタフェース上必要となるプルアップ抵抗に流れる電流は無視してよいとする．

図3.23

［3.2］ 等価回路が図3.23で表されるTTL ICの出力Yの論理式を，

（1） 入出力とも正論理の場合について求めよ．その理由も記せ．

（2） また入出力とも負論理の場合，どうなるか．

［3.3］ CMOSの特徴をTTLと比較して2点述べよ．

［3.4］ 0〜15Vのパルスで標準TTLを駆動するため，図3.24のような回路を作製したが作動しなかった．どのように変更すればよいか．

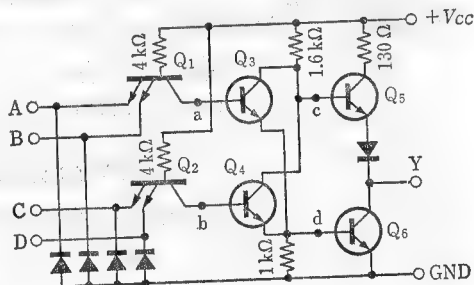
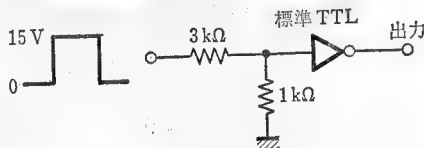



図3.24

4　汎用基本IC

ICを使用して回路を組む場合，通常は既製の汎用ICを使用する．一方，同一の回路を多量に組む場合には**セミカスタム**(半既製品)の**ゲートアレイ**や**フルカスタム**(注文品)の専用IC等を使用することが多くなる．セミカスタムと汎用の中間として，ユーザが書込み器を用いて必要論理を簡単に書き込める**PLD** (programable logic device)もある．これらを用いるとコンパクトで信頼性の高い回路を組むことができる．

本章では前章よりもより実践的に，よく使われるゲート回路を中心とする基本的汎用ICの種類や使用法について述べる．

4.1　基本ゲートIC

論理回路の基本要素がゲートICである．ゲートには**AND**，**OR**，**NOT**などがあるが，実際にゲートICとしては**NAND**，**NOR**，NOT（**インバータ**)を使用することが多い．ゲートの記号を図4.1に示す．

一般的によく使われるロジック汎用ICには**LS-TTL**（74LS～シリーズ)，**HS-CMOS**（74HC～シリーズ)，A-CMOS(74AC～シリーズ)などがある．各ゲートの速度と消費電力を図4.2に示す．また，各々の電源電圧を図4.3に示す．図4.3に示されているように，電源電圧はTTLでは5.0V ± 5% (74LS)と非常に動作範囲が狭いのに対し，CMOSは広い．

図4.4にはよく使われる基本ゲートICを示す．これらは**DIP**(dual-in-line package)と呼ばれるパッケージに入っており，そのピン配置は上から見た図

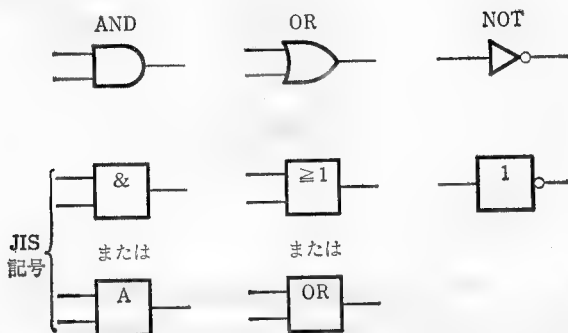



図 4.1　各種ゲートの記号

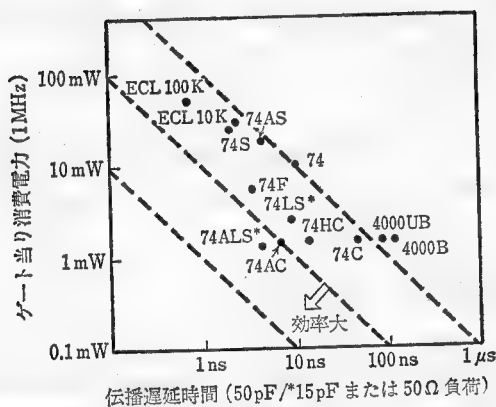



図 4.2　基本ゲートICの消費電力と伝播遅延時間（各代表値）（74および4000はそれぞれ74シリーズTTL, CMOSおよび4000シリーズCMOSを示す）

(top view)を示している．素子名は標準TTL，LS-TTL，およびHS-CMOSについて記してあるが，S-TTL等でもとくに断らない限り，機能およびピン配置は同じである．たとえばS-TTLの場合，7402は74 S 02と称され，これらはピンコンパチブル(差換え可)である．ただし，差し換えても電気的な整合性がとれるとは限らない．CMOSの場合，電源電圧は5Vとは限らない．一般にMOSでは電源ピンの呼称をV_{CC}→V_{DD}，GND→V_{SS}とすることが多い．図4.5には

図 4.3 各種ロジックファミリの動作電圧範囲

7400 (N, LS, HC, AC, …)
(a) 2入力NAND

7402 (N, LS, HC, AC, …)
(b) 2入力NOR

7404 (N, LS, HC, AC, …)
(c) NOT

7410 (N, LS, HC, …)
(d) 3入力NAND

7420 (N, LS, HC, AC, …)
(e) 4入力NAND

7430 (N, LS, HC, …)
(f) 8入力NAND

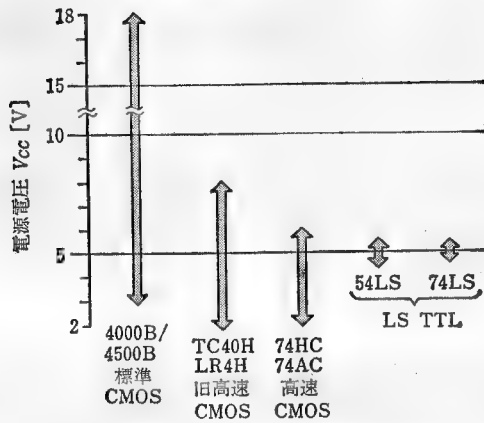
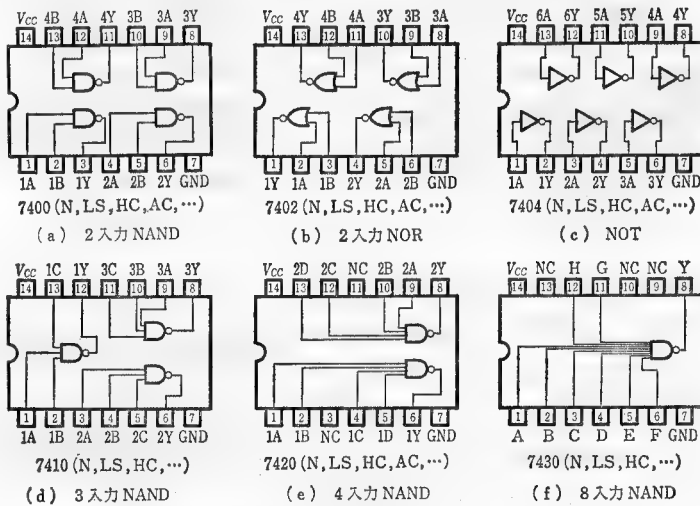



図 4.4 基本ゲート IC

ANDゲート，図4.6にはORゲート，図4.7には**Exclusive OR**（排他的論理和）ゲートを示す．さらに，大きな入力数を持つゲート素子として13入力NANDゲート74133，74HC133（図4.8）や，AND-OR形ゲートである2ワイド入力AND-ORインバータ7451，74LS51，74HC51や4ワイド入力AND-ORインバ

7408 (N, LS, HC, AC, …)

(a)　2入力AND

Vcc 1C 1Y 3C 3B 3A 3Y
14 13 12 11 10 9 8
1 2 3 4 5 6 7
1A 1B 2A 2B 2C 2Y GND

7411 (N, LS, HC, AC, …)

(b)　3入力AND

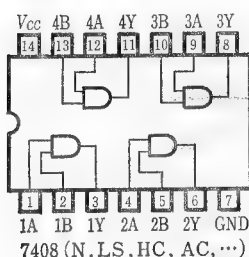
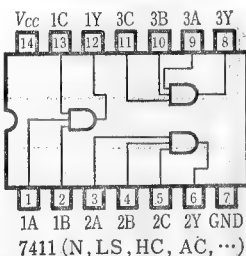
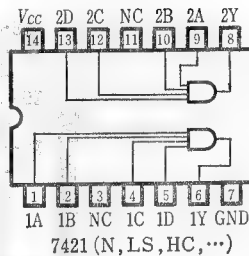
7421 (N, LS, HC, …)

(c)　4入力AND

図 4.5　ANDゲートIC

7432 (N, LS, HC, AC, …)

図 4.6　ORゲート

7486 (N, LS, HC, …)

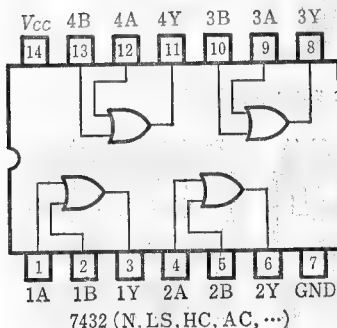
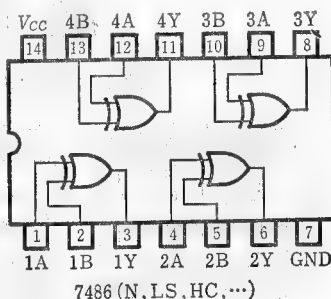



図 4.7　Exclusive ORゲートIC

ータ7454, 74LS54などもある(図4.9). これらは素子タイプによりピン配置が若干異なることもあるので注意を要する.

HS-CMOSの代表的インバータ74HC04についてその内部回路を図4.10(a)に示す. これはインバータ3段で構成されており, 1(3)段目のインバータを入(出)力バッファとみなして, **バッファ・タイプ**と呼ぶこともある. 一方, このようなバッファのない**アンバッファ・タイプ**HS-CMOSも少数ながらあり, その代表的な**74HCU04**について内部回路を図4.10(b)に示す. このようなことから, バッファタイ

74133 (LS, HC, S, …)

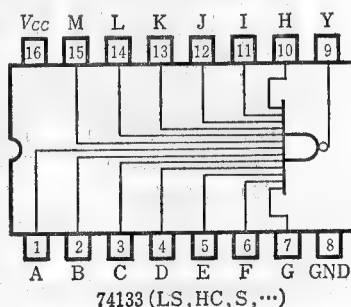



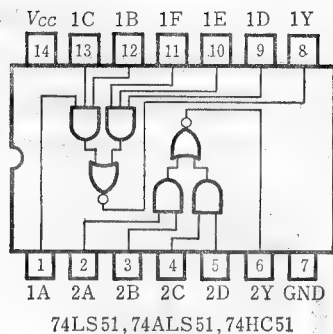
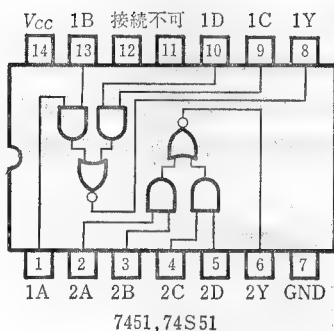
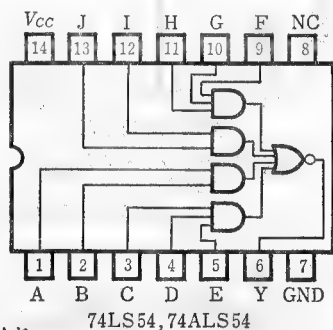
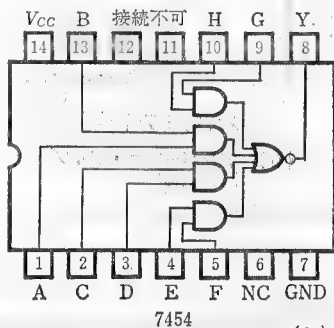
図 4.8　13入力NAND

(a) 2ワイド

(b) 4ワイド

図 4.9 2ワイドおよび4ワイド入力AND-ORインバータ

(a) バッファ・タイプ (74HC04)

(b) アンバッファ・タイプ (74HCU04)

(c) バッファ・タイプインバータのロジック図

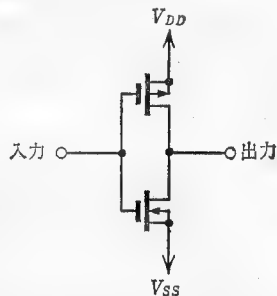
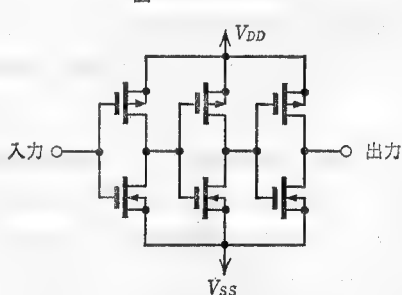



図 4.10 インバータの内部回路

プインバータの等価回路を図4.10(c)のように書くこともある.

インバータ1段の増幅度は約20倍と考えられる. したがって, バッファ・タイプとアンバッファ・タイプではその入出力特性に違いがみられる(図4.11).

(a) バッファ・タイプ
74HC04

(b) アンバッファ・タイプ
74HCU04

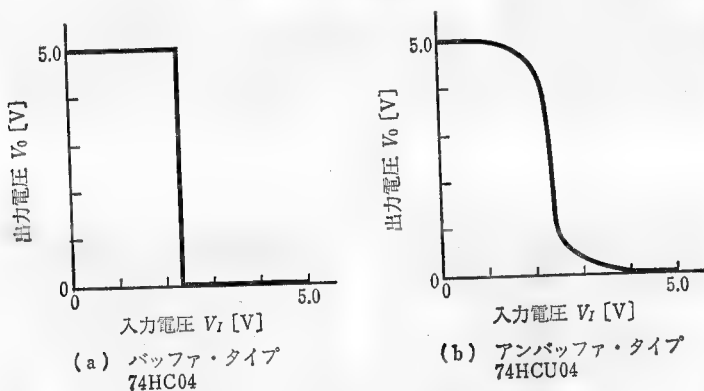



図 4.11 HS-CMOSインバータの入出力特性

4.2 未使用端子の措置および固定値入力法

ICを使用するとき, 端子 (とくに入力端子) が余ることがある. このときには, 余った入力端子はその動作に応じてHレベル, またはLレベルに固定すればよい.

未使用入力端子をHレベルに固定する場合, 74LSシリーズでは入力端子の耐圧が7Vであるため電源に直接接続する(図4.12(a)). CMOSも同様である. しかし標準TTLまたはマルチエミッタ入力端子(NANDなど)では耐圧が5.5Vであるので, 電源投入時の過渡的過電圧に耐えるよう, 図4.12(b)のように0.5〜5kΩ程度の抵抗を通して電源に接続したほうが安全である. この抵抗は10端子程度で共有してもよい. TTLにおいては入力をオープン(何も接続しない)にすると, 入力電流が流れないためH(高レベル)固定入力と同等となる(同図(c)). ただし, 後でのべるが, フリップフロップなどのトリガやリセット入力端子などは誤動作のおそれがあるため, 放置せず固定しておくほうが

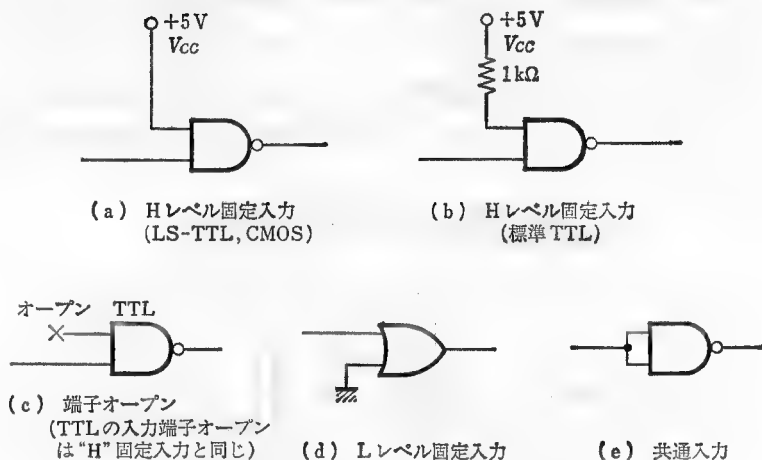



図 4.12 未使用端子の措置および固定値入力法(いずれもインバータとなる)

望ましい．いずれの素子でも入力端子をLレベルに固定する場合はGNDに直接接続すればよい(同図(d))．前段の素子の駆動力に余力があり，かつ論理的に可能ならばいずれの素子でも同図(e)のように入力端子を共通にしてよい．

CMOSにおいては入力端子をオープンにすると入力インピーダンスが高いため，雑音を拾ったり，過電流が流れたりするので，未使用入力端子は必ず電源またはGNDに固定(直接接続する)しておく．未使用ブロック

(a) 使用ブロックの措置

(b) 未使用ブロックの措置

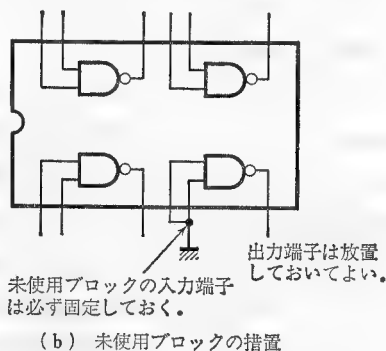
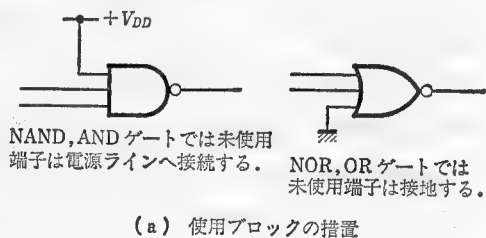



図 4.13 CMOSにおける未使用端子の措置

の入力端子も必ずすべて固定しておかなければならない．これらがTTLと異なる点である．出力端子はCMOS, TTLのいずれも放置しておいてよい(図4.13)．

4.3 シュミットトリガ回路

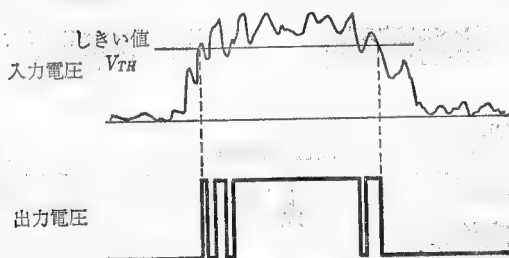



図 4.14 ヒステリシス特性のないしきい値素子

シュミットトリガは波形整形や雑音除去のために用いられる回路で，**ヒステリシス**(履歴)特性を持った**しきい**(閾)**値素子**(**いき値素子**，**2値化素子**)である．

図 4.15 シュミットトリガのヒステリシス特性(UTP：入力が下から上っていくときのしきい値電圧，LTP：入力が上から下っていくときのしきい値電圧)

図4.14のように，ヒステリシス特性を持たないしきい値素子を用いて入力波形を2値化したときの出力波形は，波形に重畳した雑音のため割れてしまう．シュミットトリガのヒステリシス特性は図4.15のように，入力信号が低レベルから高レベルへと上がっていくときはUTP (upper trip point)レベルを越えると出力は"H"となる．それ以降はしきい値が下がり，LTP (lower trip point)となる．波形の大きな落ち込み(dip)がない限り，出力は"H"に保たれる．すなわち，シュミットトリガは出力が一度"H"となれば"L"になりにくく，また一度"L"となれば"H"になりにくいようにした回路

である．これにより，雑音の重畳した波形や，ゆっくりとした立ち上りの波形の前縁，後縁で起きうる2値化出力の割れを防止し，波形整形を行う(図4.16)．

図4.17にはシュミットトリガICのいくつかを示す．図中の記号　はヒステリシス特性を持つことを示す．また，記号の向き　または　はそれに応じて反転特性または非反転特性を示すこともある．

図 4.16　シュミットトリガの出力

(a) 7414(N, LS, HC, AC, …)　(b) 74132 (N, LS, HC, …)

(c) 7413 (N, LS, HC, F)

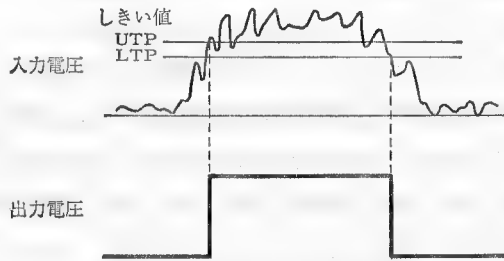
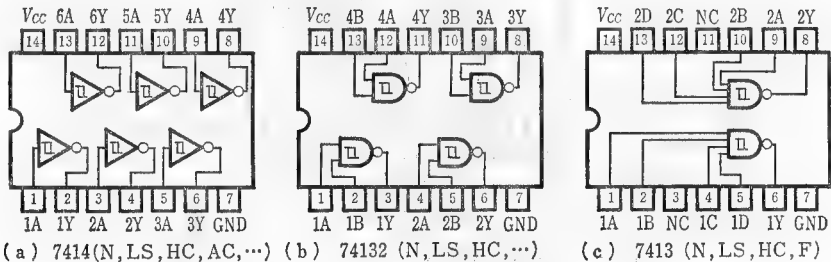



図 4.17　シュミットトリガIC

4.4　オープンコレクタとオープンドレイン

オープンコレクタ(OC)とは出力段のドライバトランジスタに内部負荷がなく，そのコレクタ端子が直接外部に出ているものである．例として，図4.18にオープンコレクタ出力NAND 74LS01を示す．＊は素子がオープンコレクタであることを示している．AND演算は基本的にD_1，D_2の2つのダイオード

(SBD)で行われる．このようにLS-TTLではマルチエミッタの代りに，**DTL** (diode transister logic)により論理演算が行われることが多い．なお，D_1，D_2以外のダイオードは保護用ダイオードである．Q_1のエミッタ電位はDTL回路の論理出力と同じ(同相)になるが，それに接続されたQ_2の出力は反対(反転出力)となる．なお，出力がLになる状態では出力端子は低インピーダンス状態であるが，出力がHになるべきときには出力端子は**高インピーダンス状態**(切離し状態)となる．

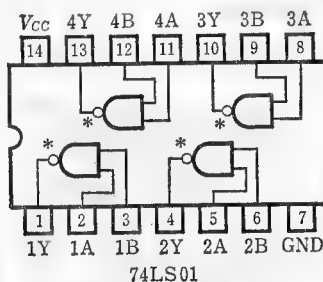
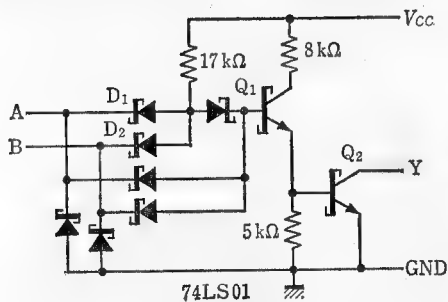
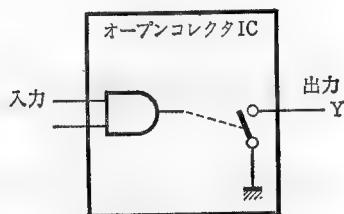
オープンコレクタにはこのほか，入力を反転せずに出力できる**バッファタイプ・オープンコレクタ**ICとして7407，74LS07(いずれも30Vまで取出し可)や7417，74LS17(同15V)があり，高耐圧であるとともに電流駆動能力(出力L時の出力電流I_{OL}はいずれも40mA，これに対し通常のTTLは16mAが標準)

(a) ピン接続

(b) 内部等価回路

(c) 出力の形式

図 4.18　オープンコレクタTTL

も大きくなっている.

図4.19にはオープンコレクタTTLの応用例として, (a)**レベル変換**, (b)**リレー駆動回路**, および(c)**LED**(発光ダイオード)**駆動回路**等を示す. これ以外に重要なものとしてワイヤードORがあるがそれは次節で述べる. (a)のレベル変換回路では, 左のTTLレベルがほぼ0〜V_{CC}のレベルに変換される. (b)の回路ではインバータタイプ・オープンコレクタTTL 7406によりリレーが駆動される. 7406の電流容量はI_{OL}=40mAであるので, それ以下の定格コイル電流のものであれば駆動可能である. なお, コイル両端の保護ダイオードはコイル電流が断になるときの逆起電力によるサージ電圧を吸収するものである. これがないとドライバトランジスタを壊すおそれがある. (c)ではオープンコレクタの出力がLとなったときLEDに電流が流れ, それが点灯する.

CMOSやHS-CMOSではオープンコレクタに代るものとして**オープンドレインIC**がつくられている. 図4.20にはその例として74HC03を示す. ただし, HS-CMOSでは使える品種は多くない. オープンドレイン回路の使用法はオー

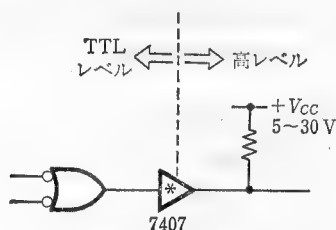
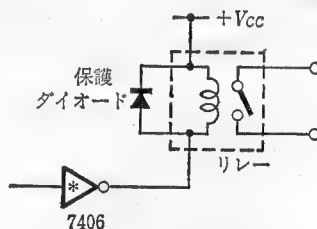
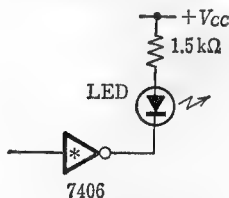
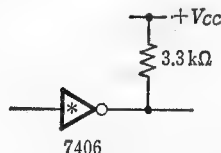



図4.19 オープンコレクタTTLの応用例

プンコレクタと同じである.

(a) ピン接続　　　　(b) 内部回路

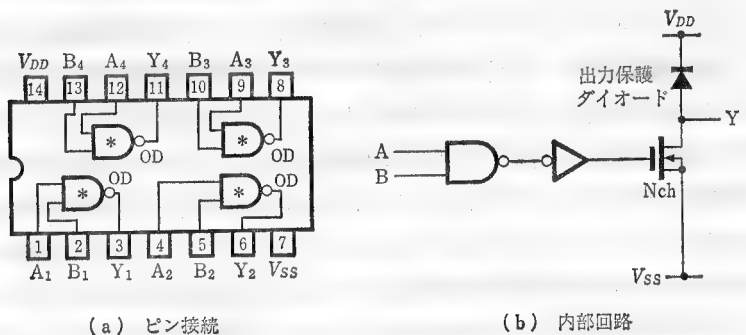



図 4.20　オープンドレイン HC-MOS 74 HC 03

4.5 ワイヤードOR

オープンコレクタやオープンドレインICを使用すると出力のORが簡単に求められる. 図4.21左に示すようにオープンコレクタの出力端子間を直接結び抵抗でプルアップしておく. これは図4.21右の回路と等価になる. オープンドレインでも同様である. このような使用法を**ワイヤードOR**という.

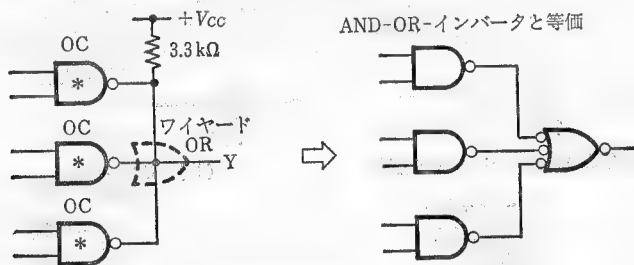



図 4.21　オープンコレクタによるワイヤードOR

4.6 トライステート出力

バスラインなど同一のラインに複数のICの出力をのせたいようなときに使

われるICの出力形式である．**トライステート出力**となっているICではH，Lのほかに**高インピーダンス状態**というのがあり，低インピーダンス状態(HまたはL)と高インピーダンス状態の切換えはコントロール端子から行う．

図4.22にトライステートICの出力の形式を示す．高インピーダンス状態とは出力端子がどこへもつながっていない状態で，出力電圧は不定である．低インピーダンス状態は通常のIC出力と同じであり，出力電圧はHレベルかLレベルのどちらかをとる．

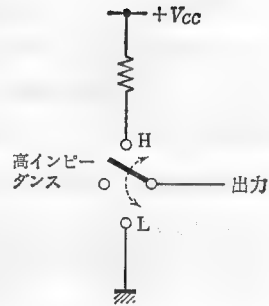



図 4.22　トライステート出力の形式

これらの出力をいくつか時間的に多重化して出力するには図4.23のように接続する．このときコントロール信号がLレベルなったICの電圧が出力ラインに出力される．この際，複数のコントロール信号が同時にL(active)にならないようにしなければならない．すべての出力が高インピーダンス状態になって出力ライン自体が高インピーダンス状態となり，出力ラインが不安定になるのを防ぐため，図に示すようにプルアップ抵抗をつけておいたほうがよい．

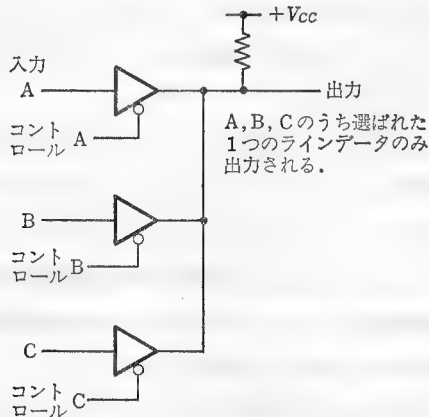



図 4.23　トライステートICによるバスラインの構成

同様の回路はトライステートICの代りに2入力オープンコレクタNANDを用いて一方をコントロール信号にしても構成できる．しかしながらオープンコレクタを使用した場合に比べトライステートICを使用すると応答時間を速くでき，かつ双方性入出力端子も駆動できる．

図4.24にトライステートIC 74125, 74126等のピン配置を示す．これらはコントロール信号のレベルが反転している．

(a) 74125 (N, LS, HC)　　(b) 74126 (N, LS, HC)

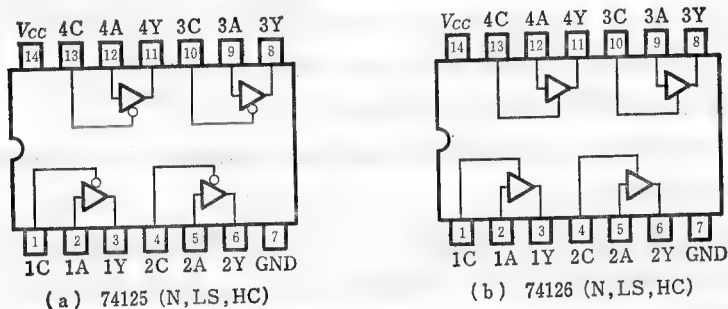



図 4.24　トライステートIC

4.7　TTLとCMOSの相互接続

TTLとCMOSを混ぜて回路を組むことがある．その際のTTLとCMOSのインタフェース法について述べる．実際に使うケースの多いCMOSを5Vで使用した場合についてのみ述べる．

図4.25にはTTLおよびHS-CMOSのロジックレベルを示す．すなわち，出力電圧はその状態“H”，“L”に応じて図に示された範囲内に必ず入り，また，入力で示された範囲内の電圧はその状態として受け入れる．その範囲外の電圧については保証されない．したがって，図中NMと記された電圧は最悪時の**ノイズマージン**(**雑音余裕度**)であり，それ以上の雑音電圧が加わると状態が反転してしまう可能性が出てくる．

(1)　CMOS→TTLインタフェース

CMOSの出力はGNDレベルからほぼ電源電圧付近まで振れるので直接TTL

(a) TTL 74～シリーズ
(N, LS, S, F, ALS, AS ;
*AS, ALS では 3.0 V)

(b) HS-CMOS 74HC～
(A-CMOS 74AC～もほぼ同じ)

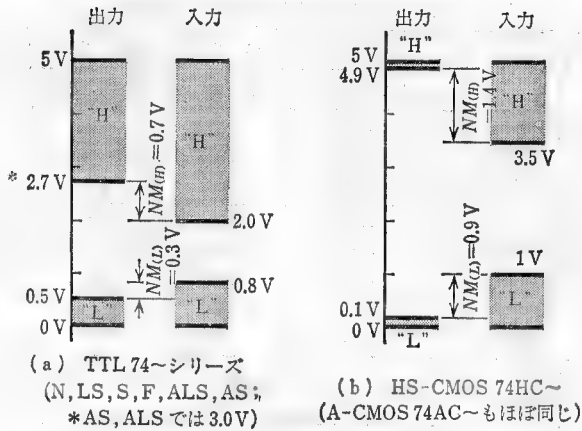



図 4.25 ロジックレベルとノイズマージン（*NM* は最悪時のノイズマージン）

の入力端子を駆動できる(図4.26(a))．表4.1に各ICの入出力電流を示す．たとえば，HS-CMOS (74HC) ではLS-TTLを10個ドライブできる．これに対し古いタイプの4000Bシリーズや40HシリーズCMOSを使用するときにはファンアウトがあまりとれないので注意を要する．

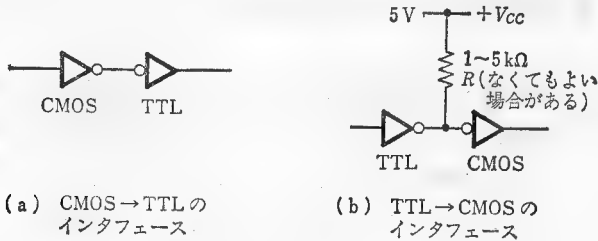
(a) CMOS→TTL のインタフェース

(b) TTL→CMOS のインタフェース

図 4.26 CMOS と TTL のインタフェース

(2) TTL → CMOS インタフェース

この場合，若干問題が出てくる．TTLの "H" レベル出力は最悪時2.7 Vと低いのでCMOSの "H" レベル入力規格(3.5～4.0V)と合わない．そのため，図4.26(b)のように，CMOSの入力端子に1～5 kΩのプルアップ抵抗を入れたほうが安全である．ただし，実際にはTTLの出力 "H"は4 V近くまで出て

表 4.1 各ロジックファミリの入出力電流代表例

		"H" レベル		"L" レベル	
		出力電流〔mA〕	入力電流〔mA〕	出力電流〔mA〕	入力電流〔mA〕
CMOS	4011B	−0.44	0.0003	0.44	−0.0003
	74 HC 00	−4.0	0.001	4.0	−0.001
	74 AC 00	−24.0	0.001	24.0	−0.001
TTL	74 LS 00	−0.4	0.02	8.0	−0.4
	74 S 00	−1.0	0.05	20	−2.0
	74 ALS 00	−0.4	0.02	8.0	−0.1
	74 AS 00	−2.0	0.02	20	−0.5
	74 F 00	−1.0	0.02	20	−0.6
ECL	10102		0.265		−0.0005
	100102		0.350		−0.0005

いることが多く，プルアップ抵抗がなくても支障のないことが多い．

HS-CMOSの場合，入力電圧や電源電圧の規格をTTLに合わせ，電気的にもTTLとコンパチブルにした**74 HCT** ○○というシリーズのHS-CMOSもあるが，品種も少なく，あまり使われていない．

4.8 単安定マルチバイブレータIC

別名**ワンショット・マルチバイブレータ**または**モノステブル・マルチバイブレータ**とも呼ばれ，入力パルス(**トリガ**)が入れば一定幅のパルスを出力するICである．出力パルスの幅は外付けのコンデンサと抵抗により決まる．使用法としてはタイミングパルスの発生などがある．

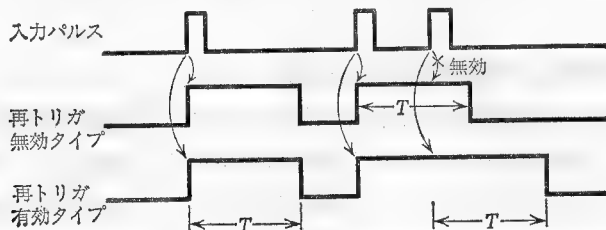



図 4.27 モノステブル・マルチバイブレータ ICの出力

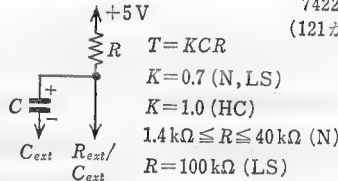
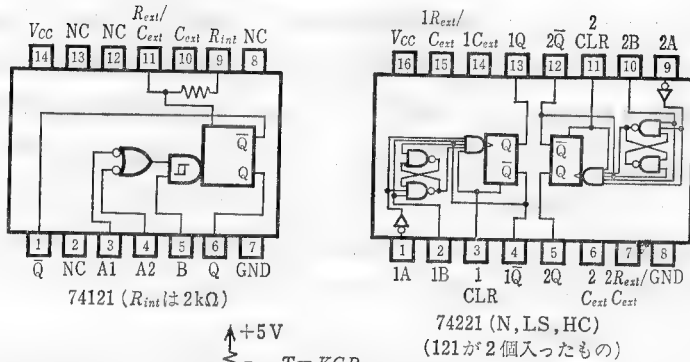
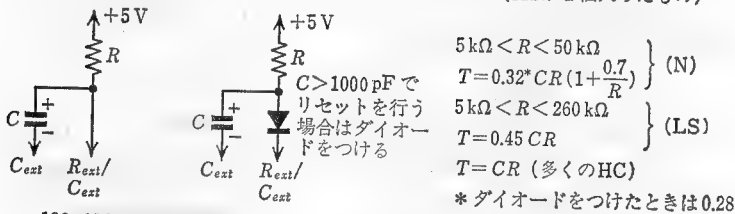
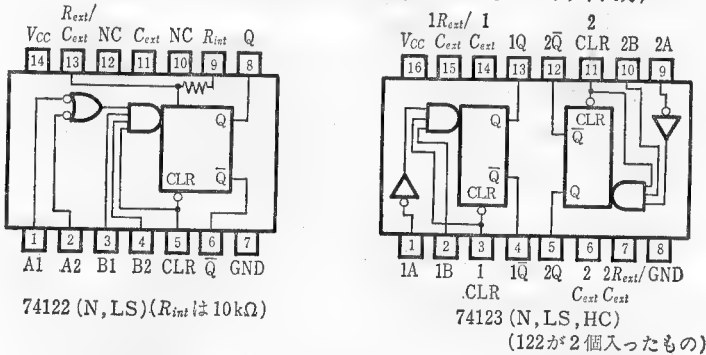



図 4.28 モノステブル・マルチバイブレータIC

このICのタイプとして，最初のトリガにより一定幅のパルスを出力中につぎのトリガが入った場合，そのトリガを無効とする(**再トリガ無効**)か，有効とするか(**再トリガ有効**)により2種のタイプがある．これらの様子を図4.27に示す．また，ICピン配置とその使用法を図4.28に示す．74121，74221等は入力にシュミットトリガ回路が入っており，221では$\overline{A}\cdot B$=Hになったときにトリガがかかる．出力パルスが出ているときにリセット信号(CLR)を入力すると，その時点で出力パルスは止まる．出力パルス幅はほぼ，

$$T=KCR$$

で与えられる．CおよびRはそれぞれ外付けコンデンサおよび外付け抵抗であり，Kは0.3〜1.0の値をとる係数である．

100 ns程度のパルスを単安定マルチバイブレータICを使わずにつくりたいときには，図4.29に示すようなワンショット回路や**論理微分回路**を使うことが

(a)　NANDによるワンショット回路

(b)　立ち上りの論理微分　　　　(c)　立ち下りの論理微分

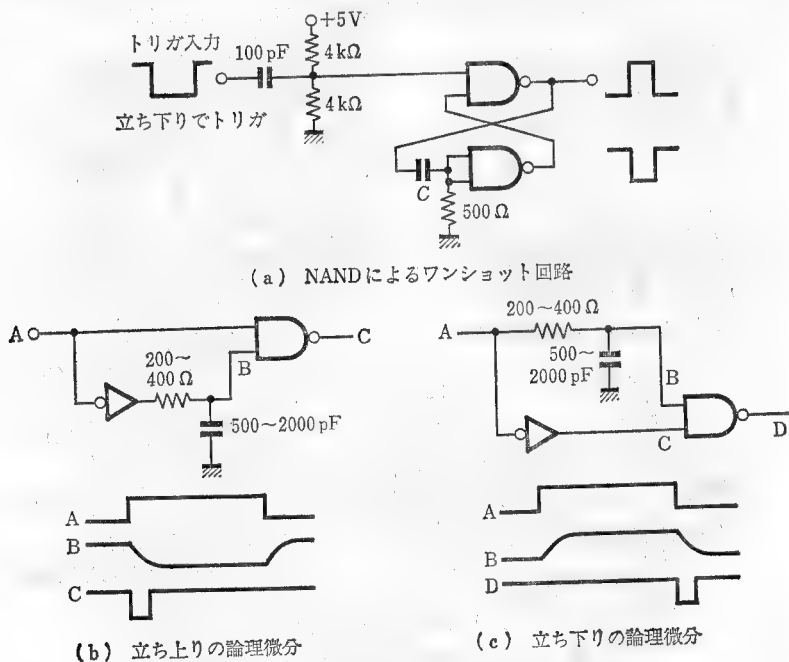



図 4.29　TTLゲートによる一定幅パルスの発生

できる．TTLを使用する場合には入力電流を流さなければならないことから抵抗の値の選択の幅が狭く，出力パルスの幅もあまり大きく変えられない．

4.9 タイマ用IC

広い範囲にわたって単安定マルチバイブレータのような一定時間幅のパルスを得たい場合や，**非安定マルチバイブレータ**（発振回路：付録A1参照）をつくりたいときに使われるものにタイマ用ICがある．代表例にNE555（バイポーラ型）とそのCMOS型であるICM7555がある．

図4.30にNE555のピン配置と内部回路を示す．図4.31には単安定マルチバイブレータとして使用するときの使用法を示す．出力電圧は電源電圧V_{CC}近くまで振れる．出力電流は200mA（NE555）と大きい．555を使用すると74121などの単安定マルチバイブレータICよりも幅の広いパルス（10s程度まで）をつくることができる．

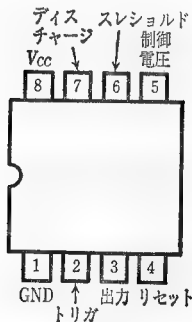
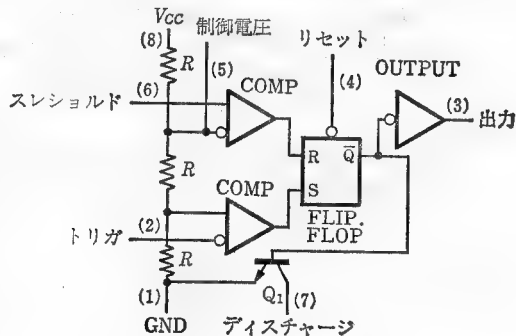
図4.32は非安定マルチバイブレータとしての使用法を示す．コンデンサCにはRを通して充電が行われる．それがおよそ$\frac{2}{3}V_{CC}$に達したらコンデンサCはショートされて放電（ディスチャージ）される．それが$\frac{1}{3}V_{CC}$位に達すると再び充電が行われる．このようにして発振が持続する．発振周波数は100kHz程

（a）ピン配置　（b）内部回路

図4.30　NE555のピン配置と内部回路

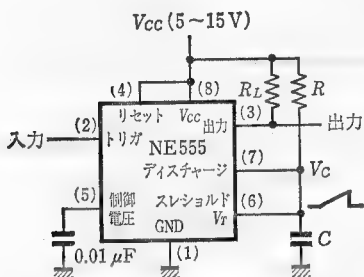
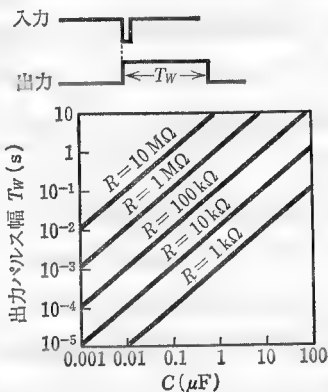
度まで可能である.

(a) 使用法　　　　(b) C, Rと出力パルス幅の関係

図 4.31　NE555によるワンショット回路と出力パルス幅（C, Rの値により出力パルス幅が決まる）

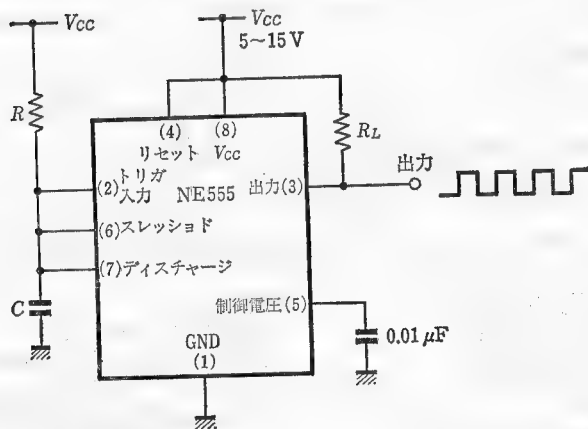



図 4.32　NE555による非安定マルチバイブレータ（C, Rの値により発振周波数が決まる）

演習問題

[4.1] つぎの語句について説明せよ.

(1) シュミットトリガ

(2) オープンコレクタ

(3) ワイヤードOR

(4) トライステート出力

[4.2] 4ビット2進→10進変換回路をゲートICにより構成せよ.

[4.3] 入力トリガパルスから任意時間遅延後，一定幅のパルスを出力する回路を単安定マルチバイブレータIC 74HC221(図4.28)およびゲートICを用いてつくれ.

5　フリップフロップ

フリップフロップ(flip-flop ; **FF**)とは反転型ゲート回路2個の各出力を相手の入力へ入れ，シーソーのように2状態“0”，“1”を交互にとり，かつその状態が出力となる素子である．

FFには状態の反転法により2種のFFがある．すなわち，

入力のレベルにより動作：RS-FF

クロックパルスにより動作：
- エッジトリガタイプFF
- レベルトリガタイプFF
- データロックアウトタイプFF

入力のレベルにより動作するFFはRS-FFのみである．一方，クロックパルスにより動作するFFにはエッジトリガタイプFF，レベルトリガタイプFF，およびデータロックアウトタイプFFがある．

5.1　RS-FF

RS-FF(または**SR-FF**ともいう)はたとえば図5.1のように構成される．真理値表を表5.1に示す．$\bar{S}(\overline{\mathrm{set}})$に0を入力すれば$Q$は1になる．$\bar{R}(\overline{\mathrm{reset}})$に0を入力すれば$\bar{Q}$が1となる．$\bar{S}$と$\bar{R}$がともに0のときには出力$Q$，$\bar{Q}$はともに1となる．なお，図5.1(a)は入力の論理的整合性を考慮しないで書いた記法であり，間違いではないが，同図(b)のようにできるだけ整合をとって記すほうが好ましい．

図5.2には，機械接点が閉じるときの**チャタリング**(振動)を防止するため使

(a) 入力論理の対応を考慮しない表現　(b) 入力論理の対応を考慮した記法

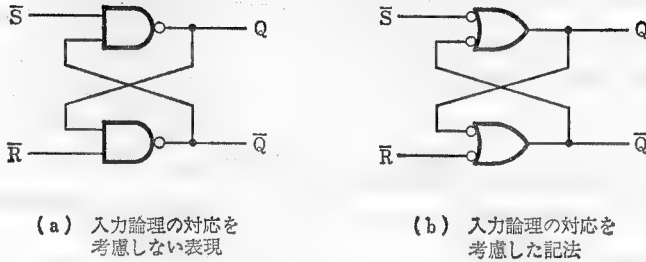



図 5.1　NANDにより構成されたRS-FF

表 5.1　RS-FF 真理値表

	入力		出力	
	$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$
①	1	1	保	持
②	0	1	1	0
③	1	0	0	1
④	0	0	1	1

*④から急に①になる場合も，必ずその途中で一瞬②または③の状態を経るので，11が保持されるのではなく，②または③の出力状態が保持される．

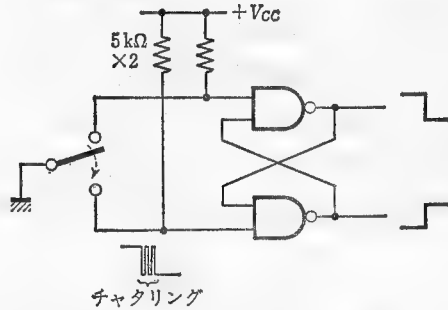
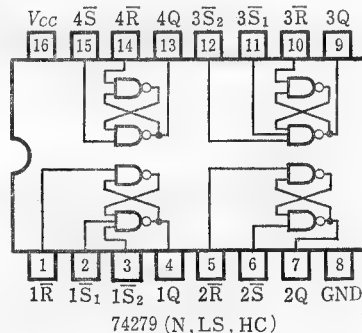



図 5.2　チャタリング防止回路

われているRS-FFを示す．誤動作を防ぐため，入力線にプルアップ抵抗がつけられている．また図5.3にはRS-FFの4個入ったIC 74279等のピン配置を示す．

図 5.3　RS-FF IC

5.2　セット優先RS-FF

RS-FFではR, Sがともに1になると出力Q, $\bar{Q}$がともに1になってしまった．

これを避けるため，セット入力またはリセット入力のどちらかを優先させることが考えられる．それに従って構成された**セット優先RS–FF**を図5.4に示す．

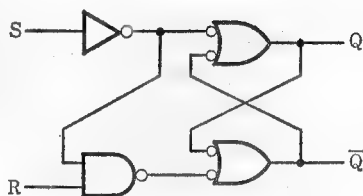



図 5.4　セット優先RS–FF

5.3　マスタスレイブFF

IC化されたFFの多くはこのマスタスレイブFF(MS–FF)である．**マスタスレイブFF**は図5.5に示すように**マスタ**と**スレイブ**の2段構えになっている．マスタは主人であり，スレイブ(奴隷)の状態はマスタの状態に従って決められる．この際，たとえば制御入力によって，またはクロックの立ち上り時の制御入力によってマスタの状態が先に決まり，つぎのクロックの立ち下り時にその状態がスレイブに移されそれが出力となる．ただし，最近のLS–TTLやHS–CMOSのFFはすべてクロックの立ち上りまたは立ち下り時のどちらか一方における制御入力により出力が決まるエッジトリガFFとなっている．

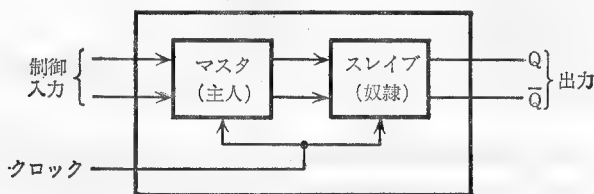



図 5.5　マスタスレイブFF

5.4　D – FF

D–FFはdelay–FFの意味である．すなわち，D端子に入力された値がクロックパルスの立ち上りエッジでD–FFに取り込まれ，出力される．IC化されたD–FF 7474等のピン配置を図5.6に，また真理値表を表5.2に示す．7474等は

7474 (N, LS, HC, AC, …)

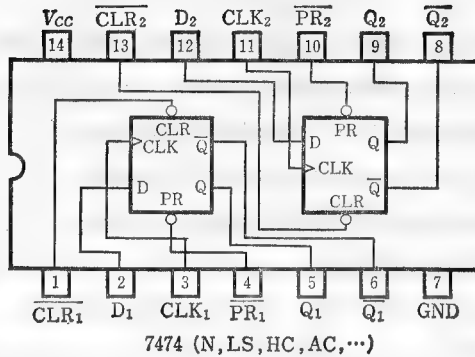



図 5.6　D-FF ピン配置

プリセット端子およびクリア端子を持っており，それらを入力するとFFの出力Qがそれぞれ1(H)および 0(L)となる．

D-FFではクロックパルスのエッジから実際にデータが出力されるまでに若干の遅れ時間がある．したがってD-FFを直接縦続接続して同一のクロックパルスでデータをつぎつぎとシフトしていく**シフトレジスタ**(8.3参照)を構成することが可能である．

D-FFの一種にデータを一時蓄える機能を持った

表 5.2　D-FF 真理値表

入	力			出	力
$\overline{PR}$	$\overline{CLR}$	CLK	D	Q	$\overline{Q}$
L	H	×	×	H	L
H	L	×	×	L	H
L	L	×	×	H*	H*
H	H	↑	H	H	L
H	H	↑	L	L	H
H	H	L	×	Q_0	$\overline{Q}_0$

注．Q_0：入力が印加される前の状態

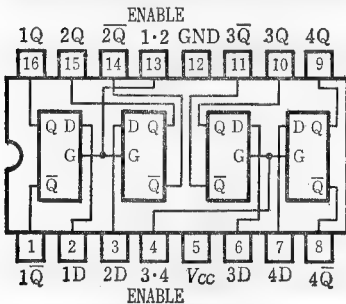
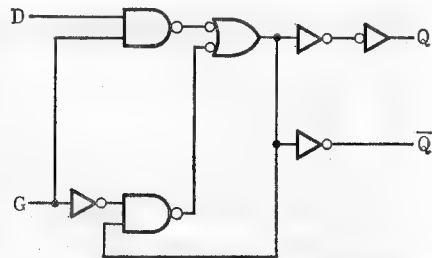
*：この状態は不安定

（a） ピン配置

（b） 内部等価回路

図 5.7　ラッチIC 7475(N, LS, HC, …)

Dラッチと呼ばれるものがある．図5.7にIC化されたラッチ7475等のピン配置およびその内部等価回路を示す．また真理値表を表5.3に示す．Gはゲートであり，それがHのとき入力データDは出力Qに“つつ抜け”状態となる．またGがLのときにはデータは保持される．したがって，GがHからLに落ちるときのデータが取り込まれ，保持される．

表 5.3　ラッチ7475等の真理値表

入力		出力	
D	G	Q	$\bar{Q}$
L	H	L	H
H	H	H	L
×	L	Q_0	$\bar{Q}_0$

×：Don't Care
Q_0：入力が印加される前の状態

5.5　JK-FF

JK-FFは，JまたはK入力のどちらか一方のみが1のときクロックに同期してその入力がFFの状態として取り込まれ，それが出力されるFFである．また，クロックパルス時にJ, Kがともに1のときにはFFの状態は反転（**トグル動作**；toggle動作）するが，逆に，J, Kがともに0のときには状態は変化しない．真理値表を表5.4に示す．

表 5.4　JK-FFの真理値表

入力		出力	
J	K	Q	$\bar{Q}$
0	0	保持	
1	0	1	0
0	1	0	1
1	1	反転	

5.6　トリガタイプ

5.3でも少し述べたが，FFのクロックによるトリガの形式には

（1）　エッジトリガ（正エッジ，負エッジ）
（2）　レベルトリガ
（3）　データロックアウト

がある．これらを図示すると図5.8のようになる．

エッジトリガ・タイプFFでは，クロックのエッジで入力データがFFに取り込まれると同時にそれが出力される．HS-CMOSやLS-TTLのFFはすべてこのタイプになっている．この動作が立ち上りのエッジ（**ポジティブエッジ**）で

行われるか，立ち下りのエッジ(**ネガティブエッジ**)で行われるか，により2つのタイプがある．ただし，クロックエッジの付近で入力データはたとえば数十nsの間一定に保たれる必要がある．エッジの直前に入力データを一定に保っておかなければならない時間が**セットアップ時間**で，エッジ直後のそれが**ホールド時間**である．エッジトリガFFを使うときクロックのエッジの変化が遅い場合には，シュミットトリガによりエッジを整形しておく必要が生じる．

レベルトリガはマスタスレイブFFにおいて使われるトリガタイプである．すなわち，クロックが立ち上り以降，入力データがマスタFFに読み込まれ，クロックが立ち下る時にその状態がスレイブFFに移され，出力される．クロックがHのときに入力データが変化すると誤動作することがある．したがって，そのような可能性のあるときにはできるだけ幅の狭いクロックパルスを用いる必要がある．レベルトリガは標準TTLのマスタスレイブFFにおいて使われるトリガタイプであるため，**マスタスレイブ型**と呼ばれることも多いが，LS-TTLやHS-CMOSのマスタスレイブ型FFにはエッジトリガしかなく，かつエッジトリガに対する意味から本書ではレベルトリガと呼ぶ．

データロックアウト・タイプはエッジトリガとレベルトリガを組み合わせたもので，クロックの立ち上りで入力データがマスタFFに取り込まれ，クロックがHのときに入力データを変化させてもその内容は変らない．そしてクロックの立ち下り時にそれがスレイブFFに移されるとともに出力される．ただし，クロックの立ち上りエッジ前後に入力データを一定に保つセットアップ時間やホールド時間が必要となってくる．標準TTLのみで使われているトリガタイ

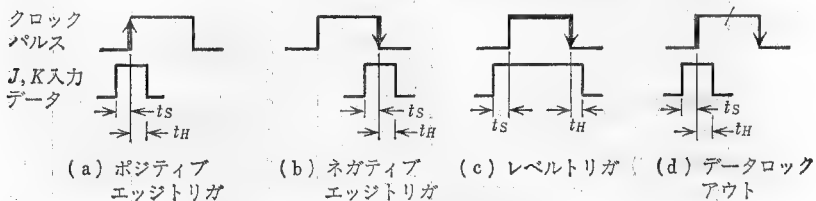



図 5.8　FFのトリガタイプ(太線はそのときの入力データがFFに取り込まれ，矢印のときに出力されることを示している．ただし，実際には，入力データは下段に示した間，一定に保っておく必要がある．t_Sはセットアップ時間，t_Hはホールド時間)

プである．

5.7　IC化されたFFのまとめ

主要なIC化FFの一覧を表5.5に示す．表中，RS-FFはクロック端子がなく，入力データのレベルにより動作する．**ラッチ**は主として数ビットのデータを並列的一時的に記憶する目的に使われ，G(enable；イネーブル(可能))端子がクロックの役目を行う．JK-FFには同一品番の標準TTLのみがレベルトリ

表 5.5　主要IC-FFの一覧

FFタイプ	型　　名	トリガタイプ	1パッケージ内回路数	1FF当りのデータ入力数	プリセット	クリア
RS-FF	74279 (N,LS,HC)	$\overline{R}$,$\overline{S}$が“L”	4	S：1,2, R：1		
D-FF	7474　(N,LS,ALS,HC,AC,*)		2	1	○	○
	74377 (LS,ALS,HC,HCT,AC,*)		8	1		
トリステートD-FF	74364 (LS)		8	1		
	74374 (LS,ALS,HC,HCT,*)		8	1		
Dラッチ	7475 (N,LS,HC,HCT)	G	4	1		
	7477 (N,LS,HC)	G	4	1		
トリステートDラッチ	74373 (LS,ALS,HC,HCT,*)	G	8	1		
JK-FF	7470 (N)		2	3	○	○
	7472 (N)		1	3	○	○
	7473 (N)		2	1		○
	7473 (LS,HC)		2	1		○
	7476 (N)		2	1	○	○
	7476 (LS,ALS,HC)		2	1	○	○
	7478 (LS,HC)		2	1	○	□
	74107 (N)		2	1		○
	74107 (LS,HC)		2	1		○
	74109 (N,LS,ALS,HC,*)		2	1	○	○
	74110 (N)		1	3	○	○
	74111 (N)		2	1	○	○
	74112 (LS,ALS,HC,HCT,*)		2	1	○	○
	74113 (LS,ALS,HC,*)		2	1	○	
	74114 (LS,ALS,HC,AC,*)		2	1	○	□

*はN(標準),LS,ALS,HC,HCT,AC以外のタイプがあることを示す．□はコモン端子．

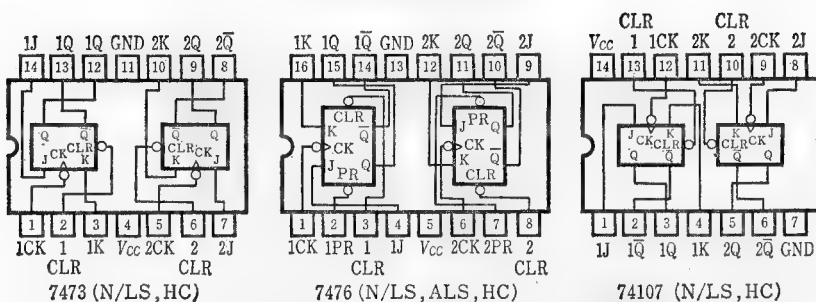



図 5.9 レベルトリガタイプ (N)とネガティブ・エッジトリガタイプ(LS, ALS, HC)のある JK-FF

表 5.6 JK-FF 7476と74LS76の真理値表

(a) 7476

入力					出力	
PRESET	CLEAR	CLOCK	J	K	Q	$\overline{Q}$
L	H	×	×	×	H	L
H	L	×	×	×	L	H
L	L	×	×	×	H*	H*
H	H	⎍	L	L	Q_0	$\overline{Q}_0$
H	H	⎍	H	L	H	L
H	H	⎍	L	H	L	H
H	H	⎍	H	H	トグル	

×はDon't Careを示す.
*は不安定であることを示す.

(b) 74LS76

入力					出力	
PRESET	CLEAR	CLOCK	J	K	Q	$\overline{Q}$
L	H	×	×	×	H	L
H	L	×	×	×	L	H
L	L	×	×	×	H*	H*
H	H	↓	L	L	Q_0	$\overline{Q}_0$
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	トグル	
H	H	H	×	×	Q_0	$\overline{Q}_0$

↓はネガティブ・エッジトリガを示す.

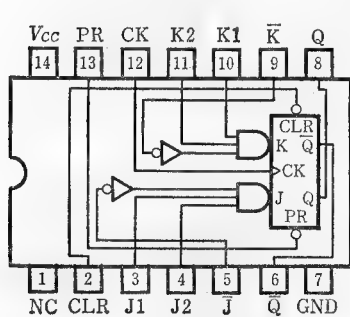
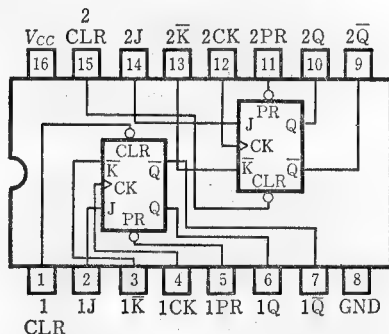



図 5.10 ポジティブ・エッジトリガタイプFF

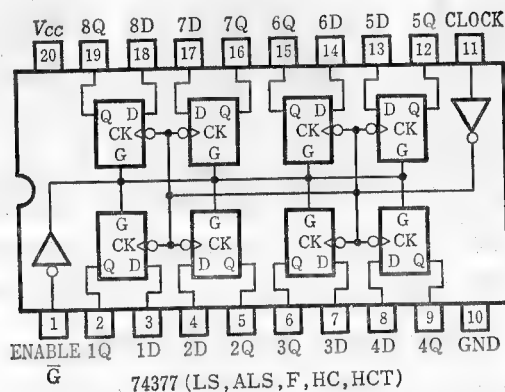



図 5.11　8ビットD-FF

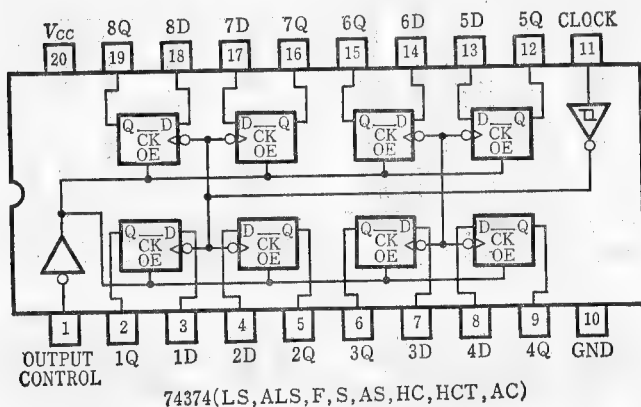



図 5.12　8ビットトライステートD-FF

ガで，LSなどそれ以外のタイプがネガティブ・エッジトリガのものがある．このようなICのピン配置を図5.9に示す．そのうちの7476，74LS76につき，真理値表を表5.6に示す．図5.10にはポジティブ・エッジトリガタイプFFのピン配置を示す．レベルトリガFFのセットアップ時間およびホールド時間はともにゼロであるが，ネガティブ・エッジトリガFFでLSタイプのものはセットアップ時間が20ns，ホールド時間はゼロとなっている．それに対し，ポジティ

ブ・エッジトリガFFでLSタイプのものはセットアップ時間の20nsに加えてホールド時間が5ns必要である．また，図5.11および5.12には8ビットD-FFを示す．さらに，図5.13にはデータロックアウトFFを示す．このうち，74110のセットアップ時間20ns，ホールド時間5nsに対し，74111ではセットアップ時間がゼロ，ホールド時間は30nsとなっている．

（a） 74110 (N)　　（b） 74111 (N)

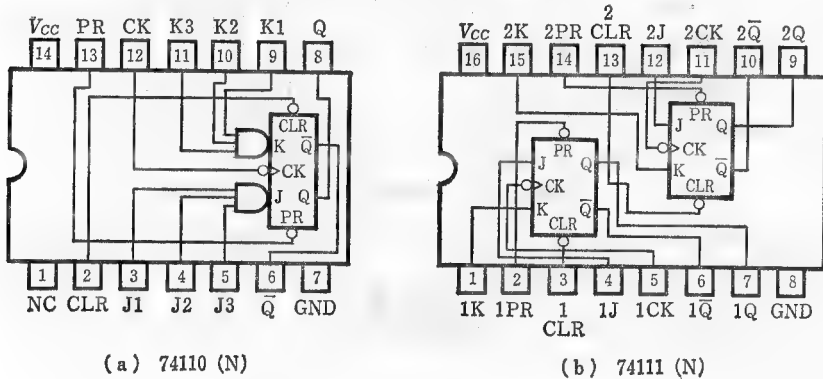



図 5.13 データロックアウトFF

5.8 各種FFの相互変換

JK-FFは図5.14(a)のように結線するとD-FFに変り得る．また，同図(b)のようにJ, K端子をともにHに固定しておくとクロック入力のたびに状態が反転(**トグル**)する．そのようなFFを**T-FF**と呼ぶことがある．

（a） JK-FFからD-FFへの変換　　（b） JK-FFからT-FFへの変換

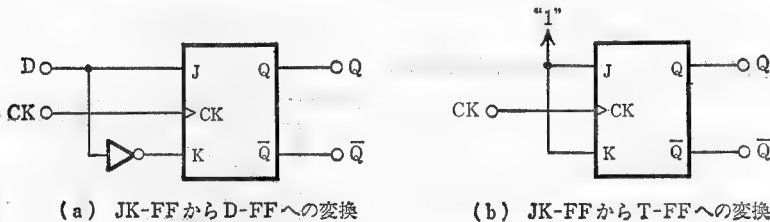



図 5.14 JK-FFの変換

一方，D-FF を JK-FF として使うためには図5.15のように結線すればよい。

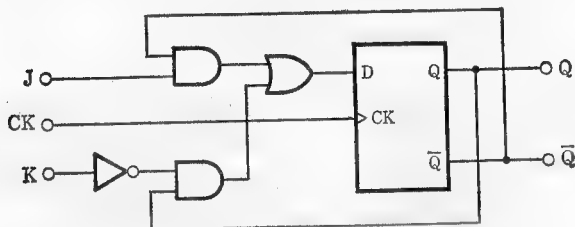



図 5.15　D-FF から JK-FF への変換

演習問題

[5.1]　つぎの語句を説明せよ．
（1）　RS-FF
（2）　マスタスレイブFF
（3）　D-FF
（4）　JK-FF

[5.2]　図5.16のような回路は状態保持機能を持つか．

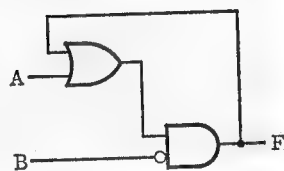



図 5.16

[5.3]　図5.17のラッチの動作を説明せよ．

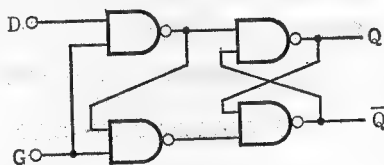



図 5.17　Dラッチ

6　ゲート MSI

6.1　コンパレータ

コンパレータは２進数の大小を比較するものである．

まず１ビットの２進数 A, B の大小問題を考えよう．この場合，表6.1に示すように，A，Bのとる値に応じて＝，＜，＞の３つの場合があることがわかる．このような機能は図6.1の回路で実現できる．またその真理値表を表6.2に示す．

多ビットになった場合，１ビットの**比較回路**を複数個用いて各ビット位置で並列的に比較を行う．最終結果は**MSB**(most significant bit；**最上位ビット**)

表 6.1　１ビット２進数の大小比較

A	B	比較結果
0	0	$A=B$
0	1	$A<B$
1	0	$A>B$
1	1	$A=B$

表 6.2　１ビット大小比較回路真理値表

入力		出力		
A	B	$A=B$	$A>B$	$A<B$
0	0	1	0	0
1	0	0	1	0
0	1	0	0	1
1	1	1	0	0

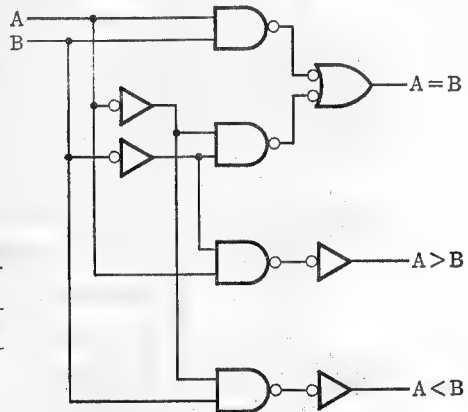



図 6.1　１ビットの大小比較回路

の比較結果を優先する．MSBの結果が＝のときのみ上から2番目のビット位置の結果を最終結果として出力する．それも等しいときは同様に上から3番目のビット位置……というようにする．このような方式に基づいてつくられた4ビット比較回路MSI 7485等のピン配置を図6.2に示す．また，その内部等価回路を図6.3に，真理値表を表6.3に示す．入力の$A>B$，$A<B$，および$A=B$は5ビット以上の比較器をつくるときの拡張用であり，下位

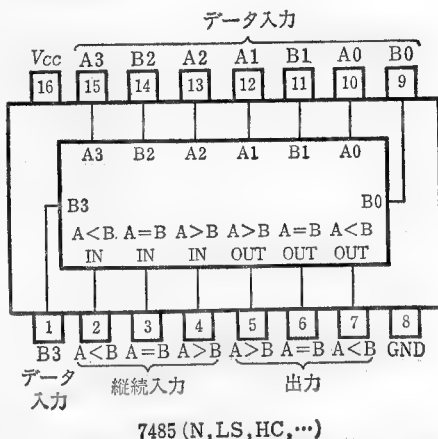



図 6.2　4ビット比較回路IC

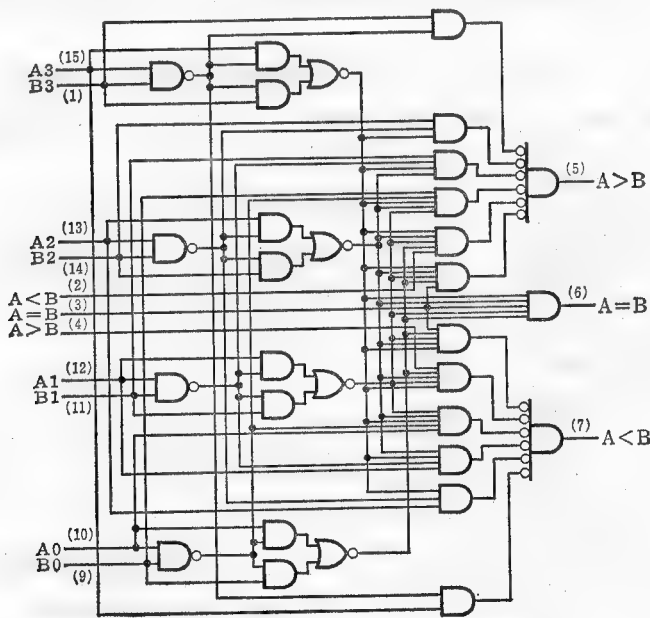



図 6.3　4ビット比較回路7485の内部等価回路

表 6.3 コンパレータ真理値表

比較入力				縦続入力			出力		
$A3, B3$	$A2, B2$	$A1, B1$	$A0, B0$	$A>B$	$A<B$	$A=B$	$A>B$	$A<B$	$A=B$
$A3>B3$	×	×	×	×	×	×	H	L	L
$A3<B3$	×	×	×	×	×	×	L	H	L
$A3=B3$	$A2>B2$	×	×	×	×	×	H	L	L
$A3=B3$	$A2<B2$	×	×	×	×	×	L	H	L
$A3=B2$	$A2=B2$	$A1>B1$	×	×	×	×	H	L	L
$A3=B3$	$A2=B2$	$A1<B1$	×	×	×	×	L	H	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0>B0$	×	×	×	H	L	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0<B0$	×	×	×	L	H	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0=B0$	H	L	L	H	L	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0=B0$	L	H	L	L	H	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0=B0$	×	×	H	L	L	H
$A3=B3$	$A2=B2$	$A1=B1$	$A0=B0$	H	H	L	L	L	L
$A3=B3$	$A2=B2$	$A1=B1$	$A0=B0$	L	L	L	H	H	L

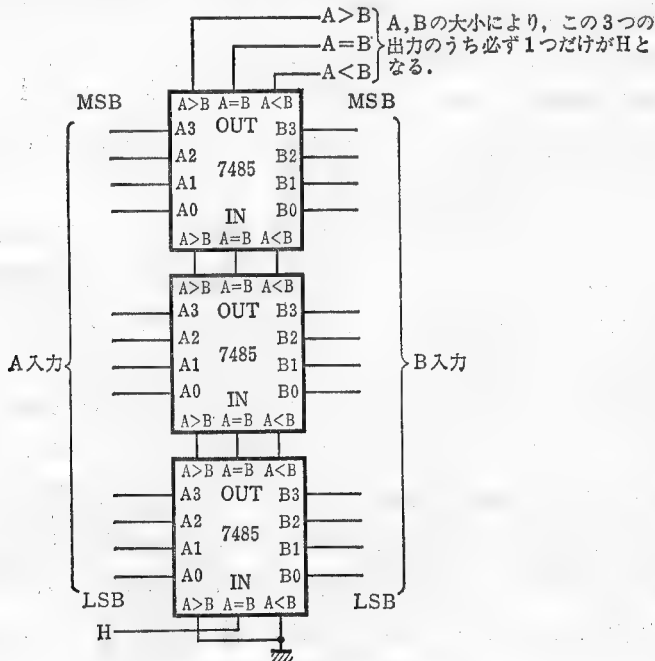



図 6.4 12ビット比較器の構成

ビット比較器の出力をここへつなぐ．使用しないときはこれらの入力端子はHに固定しておく．ただし，$A>B$ および $A<B$ はLに固定してもよい．図6.4に多ビットに拡張したときの結線図を示す．

6.2　プライオリティエンコーダ（優先順位つき符号化器）

エンコーダとは2進数に変換して出力する意味であり，プライオリティとはエンコードするときに優先順位をつけて行うという意味である．

図6.5に**プライオリティエンコーダ**の概念図を示す．左の入力端子のどれか1つのみに入力が入ってきた場合，それはそのまま2進数に直される．たとえば，入力3のみ1で他は0の場合，これは011とコード化され，右から出力される．もし2つ以上，たとえば3と6がともに入力(1)された場合，6のほうが大きいので優先され，6の2進数110が右から出力される．

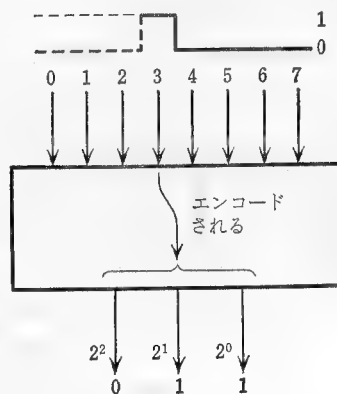



図 6.5　プライオリティエンコーダの概念

IC化されたプライオリティエンコーダのピン配置およびその内部等価回路を図6.6および6.7にそれぞれ示す．また，真理値表を表6.4および6.5に示す．

図6.6に示すように74147，74148等の入力には○印がついている．したがって，これらは入出力ともにロウアクティブ(負論理)である．表6.4の真理値表をみれば，74147の機能が理解できるであろう．すなわち，優先順位の高いほうからみていき，初めてL(○印がつけてある)になった入力の値が負論理で2進出力されていることがわかる．つぎに，表6.5をみてみよう．74148は8進数入力用プライオリティエンコーダである．まず，EIがLのときのみ結果が出力される．A0, A1, A2は結果の2進化出力である．GSは入力があること

表 6.4 74147の真理値表

入					力				出		力	
1	2	3	4	5	6	7	8	9	D	C	B	A
H	H	H	H	H	H	H	H	H	H	H	H	H
×	×	×	×	×	×	×	×	Ⓛ	L	H	H	L
×	×	×	×	×	×	×	Ⓛ	H	L	H	H	H
×	×	×	×	×	×	Ⓛ	H	H	H	L	L	L
×	×	×	×	×	Ⓛ	H	H	H	H	L	L	H
×	×	×	×	Ⓛ	H	H	H	H	H	L	H	L
×	×	×	Ⓛ	H	H	H	H	H	H	L	H	H
×	×	Ⓛ	H	H	H	H	H	H	H	H	L	L
×	Ⓛ	H	H	H	H	H	H	H	H	H	L	H
Ⓛ	H	H	H	H	H	H	H	H	H	H	H	L

○印のLがエンコードされる

表 6.5 74148の真理値表

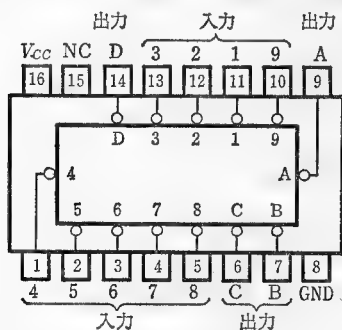
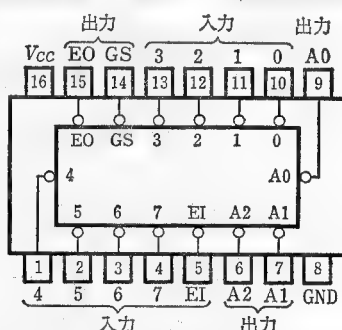
行	入					力				出			力	
No	EI	0	1	2	3	4	5	6	7	A2	A1	A0	GS	EO
1	H	×	×	×	×	×	×	×	×	H	H	H	H	H
2	L	H	H	H	H	H	H	H	H	H	H	H	H	L
3	L	×	×	×	×	×	×	×	Ⓛ	L	L	L	L	H
4	L	×	×	×	×	×	×	Ⓛ	H	L	L	H	L	H
5	L	×	×	×	×	×	Ⓛ	H	H	L	H	L	L	H
6	L	×	×	×	×	Ⓛ	H	H	H	L	H	H	L	H
7	L	×	×	×	Ⓛ	H	H	H	H	H	L	L	L	H
8	L	×	×	Ⓛ	H	H	H	H	H	H	L	H	L	H
9	L	×	Ⓛ	H	H	H	H	H	H	H	H	L	L	H
10	L	Ⓛ	H	H	H	H	H	H	H	H	H	H	L	H

(a) 74147 (N, LS, HC, …)

(b) 74148 (N, LS, HC, …)

図 6.6 プライオリティエンコーダ

を示す信号であり，EOは入力がないことを示す信号である．

図6.7の回路図をみてみよう．1の入力(L)はI1を通してG2に入っており，さらにR1を介して2^0出力であるA0として出力(L)される．しかし，もしそのとき2,4,6の入力(L)があればそちらのほうを優先するから，A0の出力は(Hに)訂正されなければならない．それゆえ，G2には2,4,6(およびEI)が入力されている．1より優先度の高い奇数入力(3.5,7)に対しては優先しても結果は変らないので考慮していない．同様に3の入力に対してはG3において，入力4または6によってマスクされるようになっている．また，2の入力に対してはG4においてI2を通ったH入力が4および5の入力でマスクされるようになっている．G1の出力EOは入力0～7(およびI3の出力)のすべてがH(入力なし)のときのみHが出力される，すなわちEIがLのとき入力が1つでもあればLが出力されるようになっている．GSはEIがLのときEOを反転した

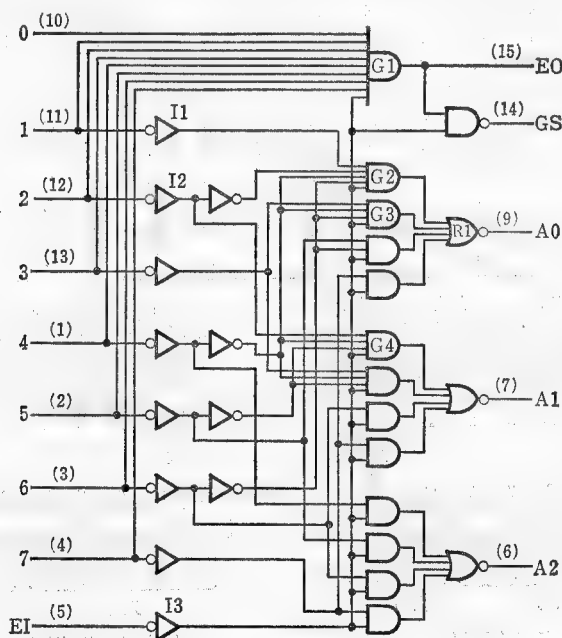



図 6.7　プライオリティエンコーダ74148の内部等価回路

ものとなっている．EIはすべての出力を出すか出さないかをコントロールする端子である．

プライオリティエンコーダの1つの応用として，デコーダと組み合わせて図6.8に示すような10進数優先回路をつくることができる．すなわち，プライオリティエンコーダ自体は0〜nの数値複数入力を優先度の高いものだけを2進数化して出力するものであるから，その出力をデコードすることにより優先度の一番高いもとの数値が出力される．

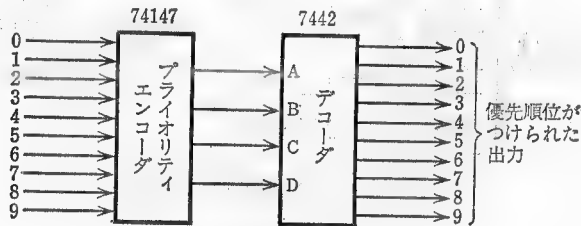



図6.8　プライオリティエンコーダとデコーダによる10進数優先回路

6.3　パリティジェネレータ

パリティ検査方式は，データに誤りがあるかどうかの検査としてよく使われている．パリティ検査方式には，**偶数パリティ検査**と**奇数パリティ検査**があるが，たとえば偶数パリティを採用した場合，データの中の1の個数が偶数になるように**検査ビット**がつけられる．それを発生させるものが**パリティジェネレータ**である．

たとえば，8ビットのデータに偶数パリティビットを1ビットつけて9ビットの偶数パリティ検査ビットつきデータにすることを考えてみよう．そのため図6.9のようにデータを2ビットずつまとめてみる．このとき，00や11の組合せは1の個数が偶数であるから偶奇の判断からは除かれる．10や01の組合せは1の個数が奇数であるから，それらの

1　0 ┆ 1　1 ┆ 0　0 ┆ 1　0

図6.9　パリティ検査ビッド生成のため，データを2ビットずつまとめる

個数を勘定しなければならない．このような機能は排他的論理で実現できる．すなわち，上のようにまとめた2ビットのデータを**排他的論理和**回路に通す．そのようにして得た排他的論理和出力をさらに2つずつまとめて排他的論理和回路に通す．このような操作を何段か(8ビット入力のときには3段，一般に 2^k ビットのときには k 段)行えば偶数パリティ出力が得られる．図6.10には8

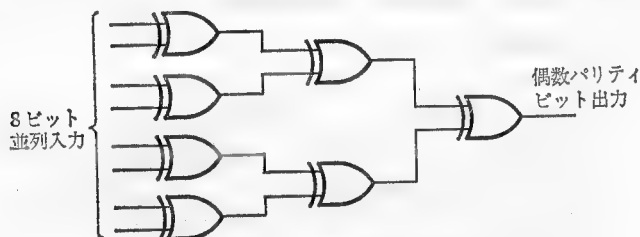



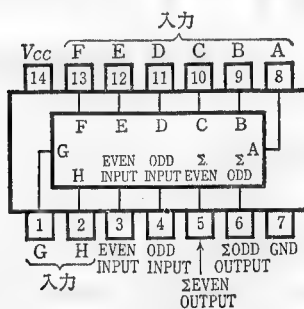
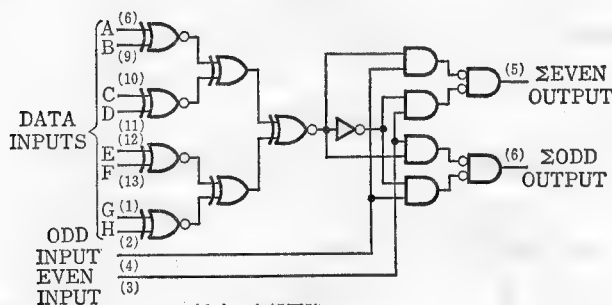
図 6.10　8ビット偶数パリティ生成回路

(a) ピン配置

(b) 内部回路

図 6.11　パリティジェネレータIC 74180 (N, S)

ビット偶数パリティビット生成回路を示す．奇数パリティ出力は偶数パリティ出力を反転すれば求まる．

8ビットデータに対するパリティジェネレータ MSI 74180 等のピン配置および内部等価回路を図6.11に示す．また，その真理値表を表6.6に示す．ここで上の4行の場合は偶，奇パリティ入力を反対レベルとした場合で，偶奇パリ

表 6.6 パリティジェネレータ 74180 の真理値表

入力			出力	
A～Hの入力のうち“H”であるものの個数	EVEN	ODD	Σ EVEN	Σ ODD
EVEN	H	L	H	L
ODD	H	L	L	H
EVEN	L	H	L	H
ODD	L	H	H	L
×	H	H	L	L
×	L	L	H	H

×：HでもLでもよい．

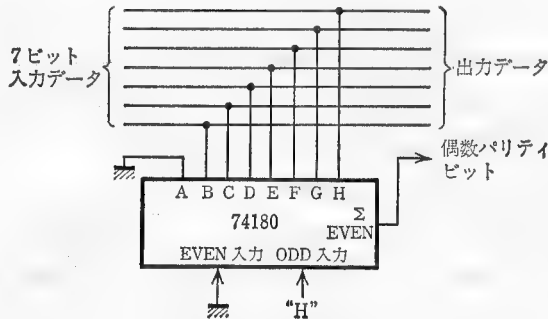



図 6.12 7ビットデータに対する偶数パリティビットの生成

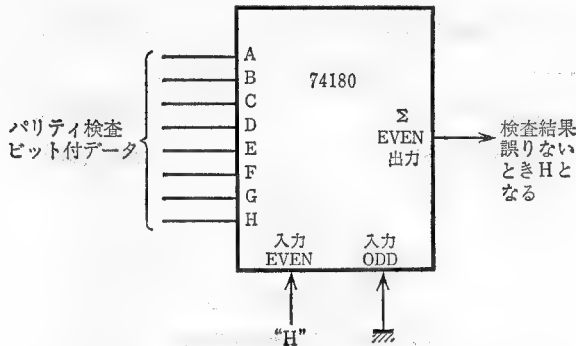



図 6.13 パリティジェネレータ 74180 によるパリティ検査

ティ出力も反対レベルで出力される。すなわち，たとえばEVEN入力をH，ODD入力をLにするとA～Hの8ビットデータ入力に対してそれらの中にHが偶数個あればΣEVEN出力がHとなる．パリティ検査ビット生成の例を図6.12に示す．なお，EVEN入力とODD入力をともにHまたはLにするとΣEVEN出力とΣODD出力がともに入力と反対のレベルになる．

パリティジェネレータはパリティ検査ビットの生成だけでなく，パリティチェックにも使うことができる．図6.13にパリティ検査への使用例を示す．

6.4　双方向性バスドライバ

マイコンなどのバスラインには入出力を双方向に使うことが多い．**双方向性バスドライバ**ICはバスの各ラインを入出力の双方向に使えるようにしたもので，入力と出力を切り換えて使う．同時に入出力はできないが，バスラインから切り離した**高インピーダンス状態**にすることはできる．

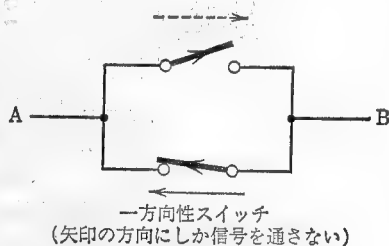



図 6.14　双方向性バスドライバ

図6.14に1本のラインのみ取り出

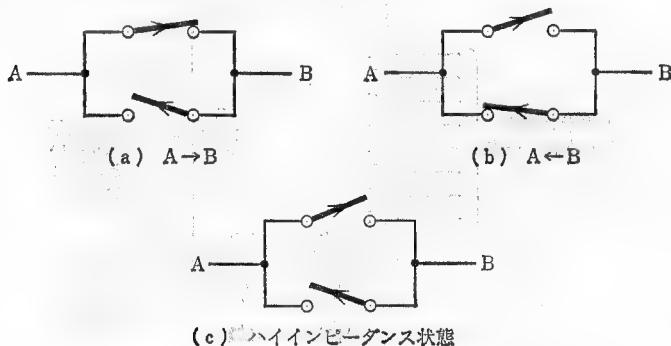



図 6.15　双方向性バスドライバの各状態

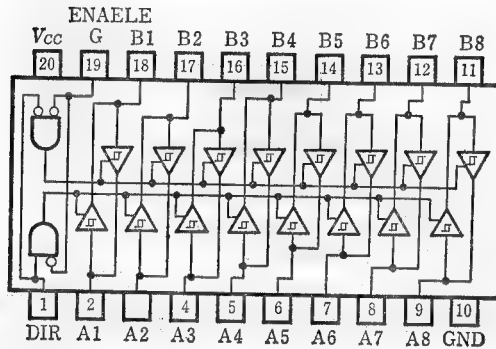



図 6.16　双方向性バスドライバ74245(LS, HC, ALS, AC, …)

した双方向性バスドライバの概念を示す．これらの各スイッチを図6.15のような各状態にすることにより，入力，出力および高インピーダンス状態とすることができる．

表 6.7　双方向性バスドライバ74 LS 245の機能表

ENABLE $\bar{G}$	DIR	動　　作
L	L	$A \leftarrow B$
L	H	$A \rightarrow B$
H	×	A, B切り離しハイインピーダンス状態

×：0でも1でもよい

図6.16には双方向性バスドライバICである74LS245のピン配置を示す．また，機能表を表6.7に示す．この表からわかるように，ENABLE $\bar{G}$がLのとき$A \rightarrow B$または$A \leftarrow B$の入出力が可能であり，$\bar{G}$をHにするとそれらは高いンピーダンス状態となり，A, B間の切り離しが行われる．また，DIRは$A \rightarrow B$または$A \leftarrow B$なる入出力方向を決める端子である．

演習問題

[6.1]　2ビットの大小コンパレータをゲートで構成せよ.

[6.2]　3進入力プライオリティエンコーダをゲートを用いてつくれ. ただし, 入力がすべて0のとき00が出力され, また, たとえばxが1のとき11が出力されるようなエンコードを行うものとする.

図 6.17　3進入力プライオリティエンコーダ

[6.3]　双方向性バスドライバについて説明せよ.

7　カウンタ構成法

カウンタはパルスの数を数えたり，時間を計ったりすることに使われ，FFを必要数用いて構成される．構成法として非同期式カウンタと同期式カウンタがある．

7.1　非同期式カウンタ

非同期式カウンタは使用する複数個のFFの同期を完全にはとらずに構成するカウンタであり，同期式カウンタより構成要素が少なくてすむ利点を持つ．

図7.1に示すように，2^n進非同期式カウンタはトグル動作(反転動作)をするFFをn個縦続接続するだけで構成できる．JK-FFの場合は，J, K端子をともにHに固定し，D-FFの場合はDを自分自身の$\overline{Q}$出力端子につなげばトグル動作をする．

非同期式カウンタのタイムチャートを図7.2に示す．各FFは前段のFF出力またはクロックパルスの立ち下り時に状態を反転させており，FFを1段経るごとにパルスの数が半分にされる(**分周**)．このことにより，入力クロックパルスの数がカウントされるのである．ただし，各段のFFは前段のFFが動作してからでないと動作しないため，将棋倒しのように時間遅れが伝播する構造となっている．なお，2^n進カウンタは0から2^n-1までカウントでき，2^n-1までカウントした状態でつぎのパルスが入ると0に戻る．電源投入時や途中でカウンタの値を0にしたいときには，リセット（またはクリア）端子つきFFを使用する．また，ある数値からカウントを始めたいときには**プリセット**（初期設

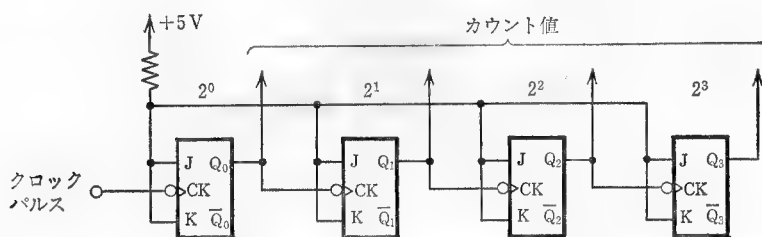
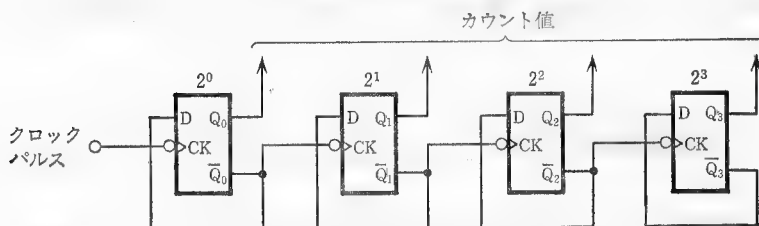
定）端子つきFFを使用する．

（a） JK-FFを用いた非同期式16進カウンタ

（b） D-FFを用いた非同期式16進カウンタ

図 7.1　16進非同期式カウンタ

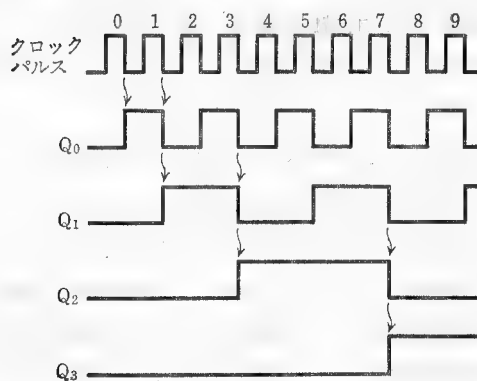



図 7.2　非同期式カウンタの動作（矢印は因果関係を表す）

7.2 N進非同期式カウンタ

2^nの形ではなく，一般のNに対して$0\sim N-1$までカウント動作をするカウンタを構成することを考えよう．まず，カウント値がNになったとき，強制的に0にリセットする方法についてのべる．これは主として非同期式カウンタで使われる方式である．

7.2.1 強制リセット法

10進カウンタを例としてとりあげる．この場合，まずFF 4段からなる16進カウンタを構成し，カウント値が10になった瞬間にカウンタをリセットすればよい．したがって，図7.3のように構成できる．図の点線のところはカウント値が10のときQの値が$0(\overline{Q}=1)$のところであり，一般にはつながなくてもよい．なぜなら，0から$N-1$までカウントする場合，各ビットとも0と1では0が先に出てくるからである．たとえば，2つの2進数

$\alpha\ \beta\ 0\ \gamma$ ………①
$\alpha\ \beta\ 1\ \gamma$ ………②
　(α, β, γは0または1)

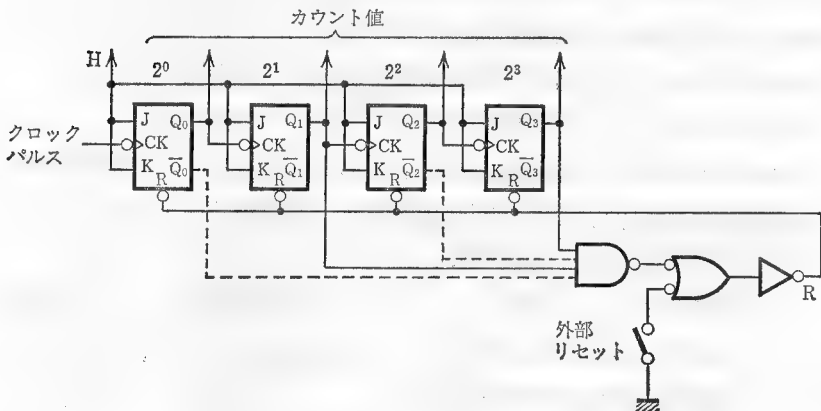



図 7.3　10進非同期式カウンタ(強制リセット法)

では上(①)のほうが必ず先に出てくるから，この場合，上から3ビット目の0はチェックすることなくα, β, γのみで10の判定ができるのである．

しかしながら，この方法はカウンタ本来のカウント範囲をオーバしたことを検出してからリセットをかけるため，瞬間的にはオーバカウント状態となり，カウンタの出力にヒゲが出ることがある(図7.4参照)．

10進数	Q_3	Q_2	Q_1	Q_0	
⋮	⋮	⋮	⋮	⋮	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	← この状態が瞬間的に入るため，Q_1の出力にヒゲが入る．
0	0	0	0	0	

図7.4　10進非同期式カウンタの出力

7.2.2　修　正　法

強制リセット法によるN進非同期式カウンタは構成が簡単明快であるが，出力にヒゲが出る欠点があった．このようなヒゲの生じないN進非同期式カウンタの構成法として，状態が$N-1$になったとき，つぎの(N個目の)クロックパルスが初段FFに入るのを止め，そのクロックパルスが立ち下ったときリセットパルスを出すようにする方法が考えられる．考え方は簡単であるが，実際にこの方法で構成すると複雑になってしまうので，代りにN個目のクロックパルスが入ったとき，各段のFFが状態$N-1$から0に遷移するようにクロックパルスを制御することを考える．

10進カウンタ($N=10$)の場合を考える．表7.1に$N-1$から0に遷移するときの状態の変化を示す．これより状態の変化としては$1 \to 0$と$0 \to 0$があることがわかる．FF_iの出力をQ_iとする．

表7.1　10進非同期式カウンタの状態遷移

カウント値 ＼ Q_i	Q_0	Q_1	Q_2	Q_3
9(N−1)	1	0	0	1
0	0	0	0	0

(1)　Q_iが$1 \to 0$となる場合

(1-a)　前段のQ_{i-1}すなわちFF_iの入力が$1 \to 0$であればFF_iの入力は修正の必要がない．なぜなら，

Q_{i-1}の変化により，Q_iは$1\to0$となるからである．初段のFF_0についてはQ$_{i-1}$はクロックパルスであると考えればよい．すなわち，クロックパルスは必ず$1\to0$と変化すると考えて入力の修正は必要ないと考える．

(1-b)　前段のQ_{i-1}が$0\to0$であればQ_iは変化しないのでクロックに合わせてなんらかの方法でFF_iを反転させてやる必要がある．そこで，状態$N-1$のとき1となっているFFの出力のすべてとクロックパルスのANDをとり，これと前段の出力Q_{i-1}のORをとったものをFF_iのクロック入力とする．このようにするとFF_iに強制的に外部からクロックパルスが挿入される形となり，Q_iは0となる．

(2)　Q_iが$0\to0$となる場合

(2-a)　前段のQ_{i-1}が，すなわちFF_iの入力が$1\to0$であればそのままにしておくとQ_iは$0\to1$となってしまうので，このFF_iがクロックパルスによって変化しないように，カウンタの状態が$N-1$になったとき0となる信号(FF_iより上位のFFのうち1が立っているもののNAND出力でよい)をJ_i(FF_iのJ入力端子)に入れておく．初段のFF_0の場合

表 7.2　10進非同期式カウンタの入力修正法

カウント値 ＼ Q_i	Q_0	Q_1	Q_2	Q_3
9	1	0	0	1
0	0	0	0	0
修正措置	(1−a)により不要	(2−a)	(2−b)により不要	(1−b)

図 7.5　10進非同期式カウンタ

にはクロックパルスをQ_{i-1}と考えてこの措置をしておく．

(2-b)　Q_{i-1}が0→0であればQ_iはそのまま0を保つので何も措置をしない．

非同期式10進カウンタの場合にこれらの方式に従ってとられる修正措置を表7.2に示す．また回路構成を図7.5に示す．

7.3　*N*進同期式カウンタ

各段のFFがつぎにとるべき状態をもとにカルノー図を用いて回路を構成する方法について述べる．例として，6進カウンタをとりあげる．6進カウンタは$2^2<6\leq 2^3$であるから3段のFFで構成できる．各FFの状態を**LSB**(最下位ビット)よりQ_0, Q_1, Q_2とする．そして，各カウンタがつぎにとるべき状態(カウント値が1増えたときの状態)を各々Q_0', Q_1', Q_2'とする．カウンタのカウント値は0から5まで順次増加した後，0に戻る．したがって，カウンタの現在の状態とつぎにとるべき状態を求めると表7.3のようになる．そこで，各々のつぎの状態(Q_0', Q_1', Q_2')についてカルノー図をつくると表7.4のようになる．

表 7.3　6進カウンタのつぎにとるべき状態

状態＼カウント値	現在の状態 Q_0	Q_1	Q_2	つぎの状態 Q_0'	Q_1'	Q_2'
0	0	0	0	1	0	0
1	1	0	0	0	1	0
2	0	1	0	1	1	0
3	1	1	0	0	0	1
4	0	0	1	1	0	1
5	1	0	1	0	0	0

表 7.4　6進カウンタのカルノー図

Q_0'

Q_0Q_1＼Q_2	0	1
00	1	1
10	0	0
11	0	×
01	1	×

Q_1'

Q_0Q_1＼Q_2	0	1
00	0	0
10	1	0
11	0	×
01	1	×

Q_2'

Q_0Q_1＼Q_2	0	1
00	0	1
10	0	0
11	1	×
01	0	×

図 7.6 6進同期式カウンタ

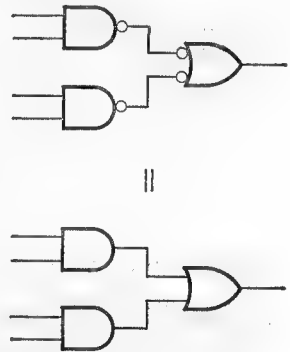
図 7.7 上と下は同じ機能を持つ

これらのカルノー図の簡約化より，つぎにとるべき状態はつぎのようになる．

$$Q_0' = \overline{Q}_0 \qquad (7.1)$$

$$Q_1' = \overline{Q}_0 Q_1 + Q_0 \overline{Q}_1 \overline{Q}_2 \qquad (7.2)$$

$$Q_2' = Q_0 Q_1 + \overline{Q}_0 Q_2 \qquad (7.3)$$

これらの式より，6進カウンタは図7.6のようにして構成できる．カウント値は各FFのQ_n($n=0, 1, 2$)より取り出す．なお，ここで図7.7の上の回路と下の回路は同じ機能を持つことに注意しよう．

7.4 非同期式減算カウンタ

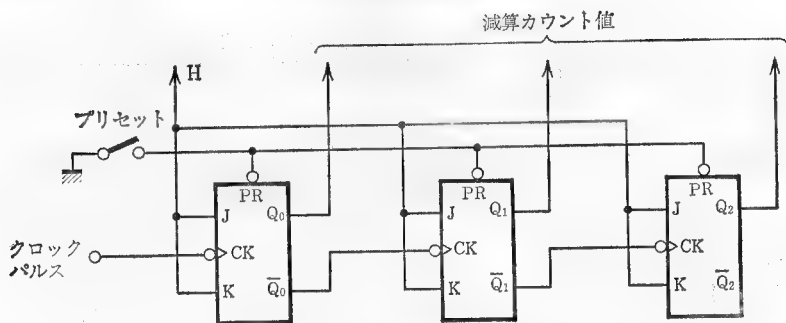
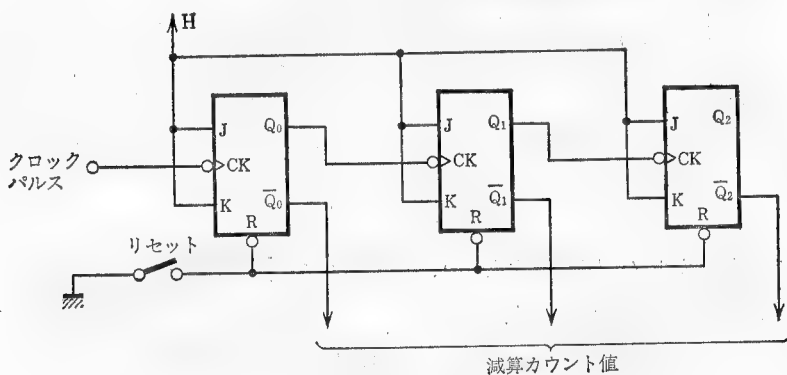
上に述べたカウンタはカウント値が1ずつ増えていくものであった．逆に1

（a）次段クロックを$\overline{Q}_n$から取る方法

（b）出力を$\overline{Q}_n$から取る方法

図 7.8　減算カウンタ

ずつ減少していくカウンタもある．減算カウンタの使い方として，最初に一定の初期値をFFに設定し，順次カウント値を減らしていき，カウント値が0になれば所定のカウント値になったということでカウント終了とする使い方もよく行われる．

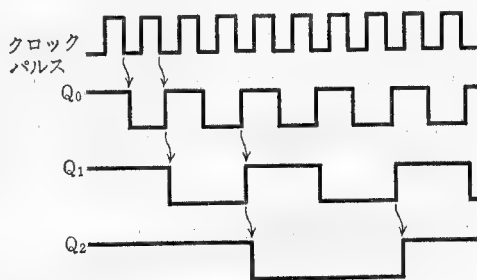



図 7.9　減算カウンタタイムチャート

減算カウンタ(ダウンカウンタ)は通常の加算カウンタにおいて，次段に接続する出力をQ_nの代りに$\overline{Q}_n$から接続すればよい(図7.8(a))．なお，カウンタ出力をQ_nの代りに$\overline{Q}_n$からとることによっても減算カウンタになる(図7.8(b))．減算カウンタのタイムチャートを図7.9に示す．これらはいずれも，初期値111_2から減算していくようになっている．

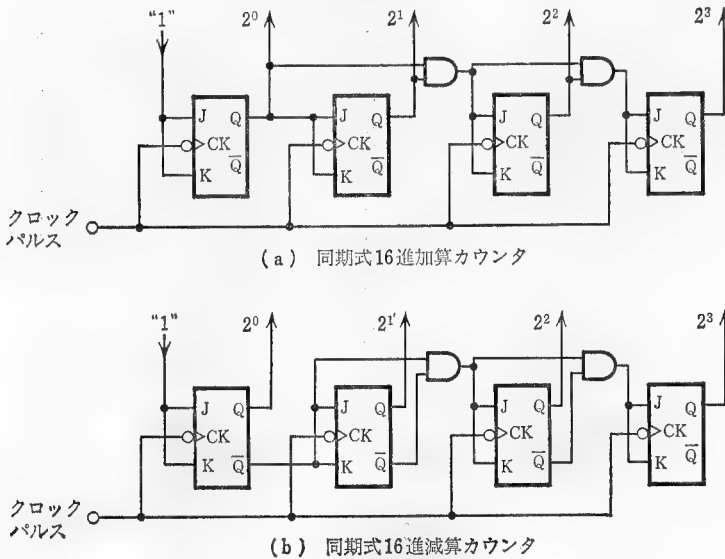
7.5 同期式減算カウンタ

上と同様の考え方で，次段入力のためのANDをQの代りに$\overline{Q}$からとれば減算カウンタが実現できる．図7.10に16進加算カウンタと16進減算カウンタの比較を示す．なお，非同期式減算カウンタと同じく，カウンタ出力をQ_nの代りに$\overline{Q}_n$からとることによっても減算カウンタとすることができる．

(a) 同期式16進加算カウンタ

(b) 同期式16進減算カウンタ

図 7.10 同期式カウンタ

（a）同期式リップルキャリー形可逆カウンタ（低速形）

（b）同期式高速形可逆カウンタ

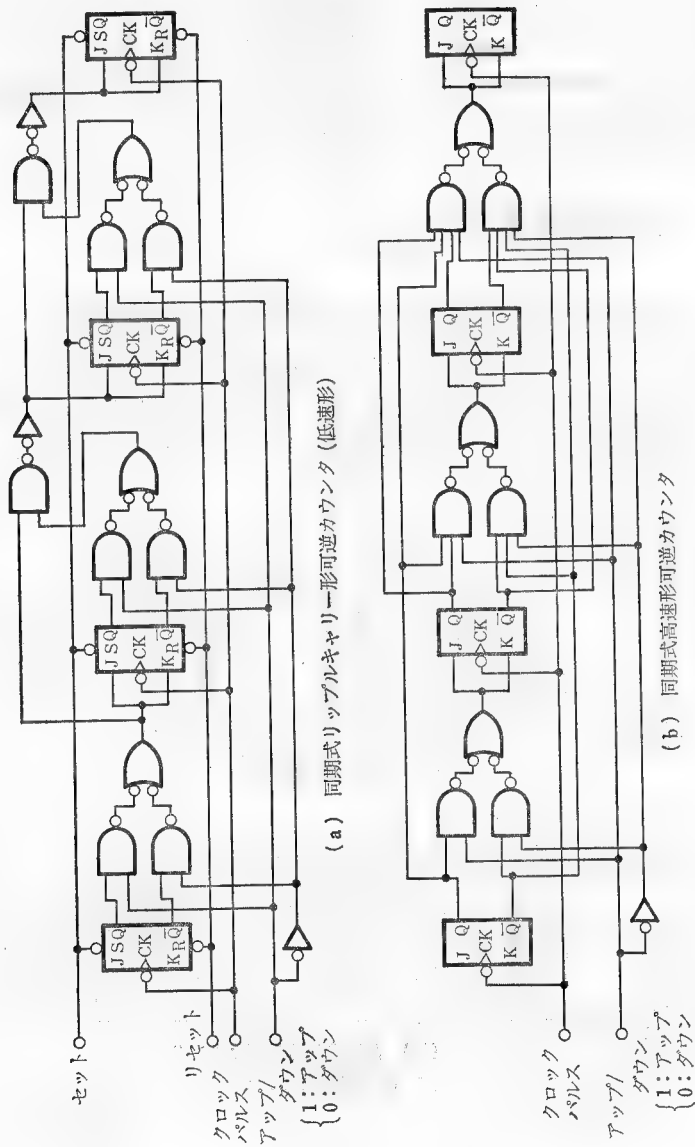



図 7.11　同期式可逆カウンタ

7.6 可逆カウンタ

加算，減算の両方の機能を持ったカウンタで，制御入力により加算も減算もできる．したがって，たとえば0〜9まで加算していき，その後0まで減算していくことなどができる．

先に減算カウンタの節で述べたことからわかるように，可逆カウンタは図7.11のように，次段FFへの入力を制御入力で切り換えることで実現できる．非同期式でこのような切換えを行うとアップ／ダウン切換え入力の影響でFFがトリガ(状態反転)され，誤動作する．そこで，ICされたカウンタではこのような問題をさけるためにもすべて同期式が使われているのである．

演習問題

[7.1] 非同期式カウンタと同期式カウンタの違いについて述べよ．

[7.2] JK-FF 74LS76(図7.12)およびゲート類を用いて，7進非同期式カウンタをつくれ．

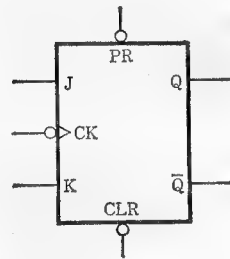



図 7.12 JK-FF 74LS76

[7.3] D-FF 7474(図7.13)およびゲート類を用いて5進同期式カウンタをつくれ．

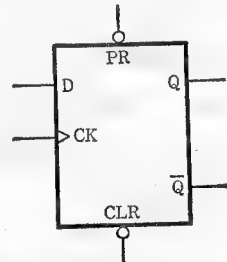



図 7.13 D-FF 7474(N, LS, HC, AC)

8 カウンタ MSI

8.1 同期式カウンタ MSI

IC化されたカウンタは安定性の面から，すべて同期式である．**同期式カウンタ**MSIには4ビットの10進カウンタと16進カウンタがある．

10進カウンタには74160(N, LS, HC, …)，74162(N, LS, HC, …)があり，16進カウンタには74161(N, LS, HC, …)，74163(N, LS, HC, …)がある．これらはクロックの立ち上りでトリガされ，カウント動作が行われる．また，任意の初期値を並列にセットできるようになっている．

図8.1には4ビット同期式カウンタMSIのピン配置および内部回路を示す．また，動作表を表8.1に示す．74160はクリア端子が各FFのクリア入力に接続されており，クリア信号が入ればカウンタ出力は他に優先して随時クリアされる．これを，**非同期クリア**という．それに対し，初期値のロード(**プリセット**)は，あらかじめデータ入力端子に用意されたデータ(初期値)がクロックのポジティブエッジ(立ち上り)で各段のFFに一斉にとり込まれるようになっている．これを**同期プリセット**という．74160(10進)および74161(16進)は非同期クリアであり，74162(10進)および74163(16進)は**同期クリア**(クロックのポジティブエッジでクリアされる)である．なお，プリセットに関してはいずれも同期プリセットである．

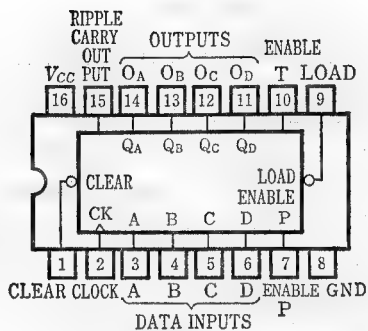
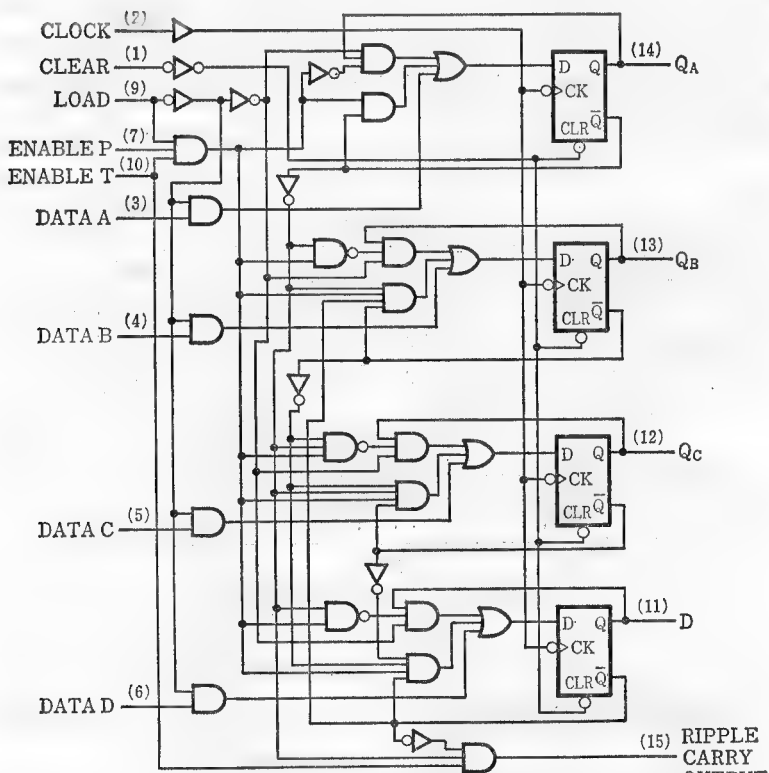
イネーブルPおよびイネーブルTはいずれも“H”としなければカウント動作をしない．**リプルキャリー**出力は10進カウンタの場合はカウント値“9”の期

（a） 74160（10進），161（16進），162（10進），163（16進）のピン配置

（b） 74LS160A（10進）の内部回路

図 8.1　4ビット同期式カウンタ MSI(74160(N, LS, HC, …)，74162(N, LS, HC, …)，74163(N, LS, HC, …))

表 8.1　4ビット同期式カウンタ 74160 の動作状態

入力					出力		動作
Clear	Load	CK	Enable P	Enable T	$Q_AQ_BQ_CQ_D$	Ripple Carry	
H	H	↑	H	H		―	カウント
H	L	↑	×	×	$D_AD_BD_CD_D$	―	データセット
L	×	×	×	×	L L L L	―	クリア
H	×	×	×	H	H L L H	H	―

間中ずっと，また16進カウンタの場合にはカウント値“15”の期間中ずっと出る．同期式カウンタを複数個縦続接続して4ビット以上のカウンタを構成するときには，図8.2のように，リプルキャリー出力を次段のイネーブルP, T端子に接続する．

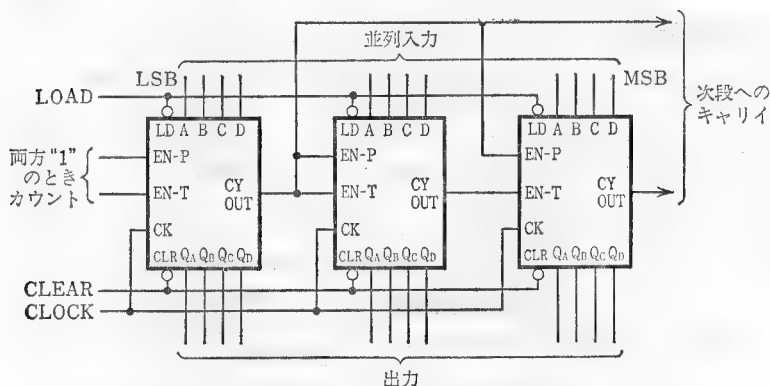



図 8.2　同期式カウンタ MSI の接続(高速動作)

8.2　可逆カウンタ MSI

同期式(加算)カウンタと同じく，同期式可逆カウンタ MSI にも4ビット10進カウンタと16進カウンタがある．10進カウンタには，74168(LS, S, ALS,

…)，74190(N, LS, HC, …)，74192(N, LS, HC, …)があり，16進カウンタには74169(LS, S, ALS, …)，74191(N, LS, HC, …)，74193(N, LS, HC, …)がある．同期式カウンタと同じく，これらはいずれもクロックのポジティブエッジでカウント動作を行う．

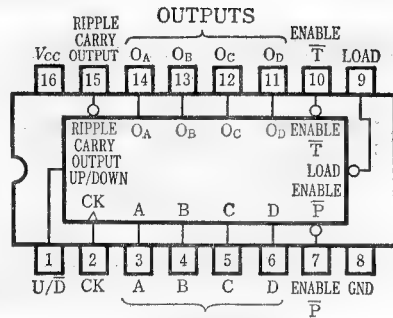
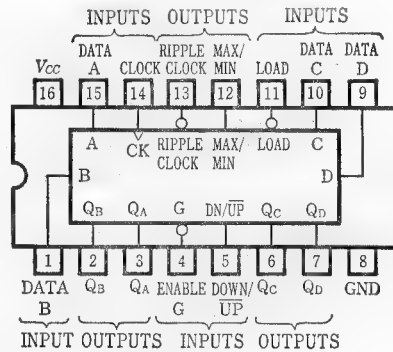
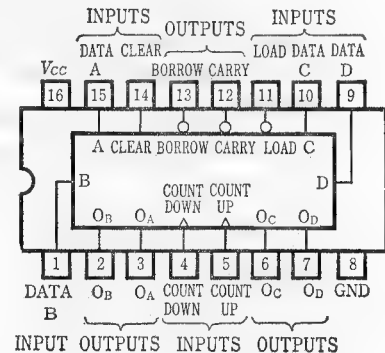
図8.3に可逆カウンタMSIのピン配置を示す．また，多ビットカウンタへの拡張法を図8.4に示す．表8.2には74LS168の動作表を示す．168, 169, 190, 191においては加算，減算の切換えはup/down入力端子(U/$\overline{\text{D}}$またはD/$\overline{\text{U}}$)をアップ(U)にするかダウン(D)にするかにより行う．クロック入力はアップとダウンで共通である．168, 169の多ビット拡張法はU/$\overline{\text{D}}$端子がある点を除けば同期式カウンタのそれと同じである．190, 191はリプルキャリーの代りに**リプルクロック**が出てくるため，図8.4(b)のように接続することにより多ビットへの拡張を行うことができる．これに対して，192, 193ではクロック入力端子がアップとダウンで別々になっており，図8.4(c)のように拡張接続される．データの

(a) 74168 (LS, S, ALS, …)
74169 (LS, S, ALS, …)

(b) 74190 (N, LS, HC, …)
74191 (N, LS, HC, …)

(c) 74192 (N, LS, HC, …)

図 8.3 可逆カウンタのピン配置

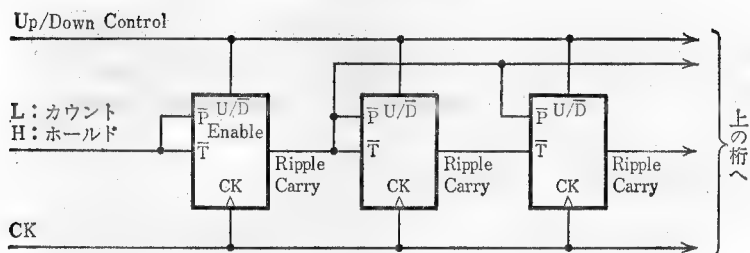
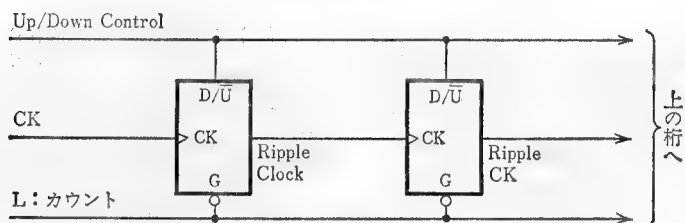
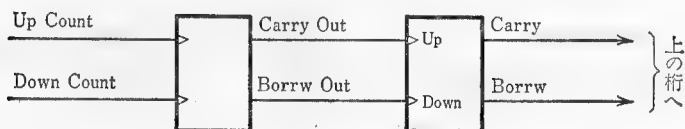



図 8.4　可逆カウンタの多ビットカウンタへの拡張

表 8.2　同期式プリセット可能10進カウンタ74LS168の動作状態

入力					出力		動作
Load	$U/\overline{D}$	CK	Enable $\overline{P}$	Enable $\overline{T}$	$Q_AQ_BQ_CQ_D$	Ripple Carry	
H	H	↑	L	L	—	—	カウントup
H	L		L	L	—	—	カウントdn
L	×		×	×	$D_AD_BQ_CQ_D$	—	データセット
×	H	×	×	L	H L L H	L	—
×	L	×	×	L	L L L L		—

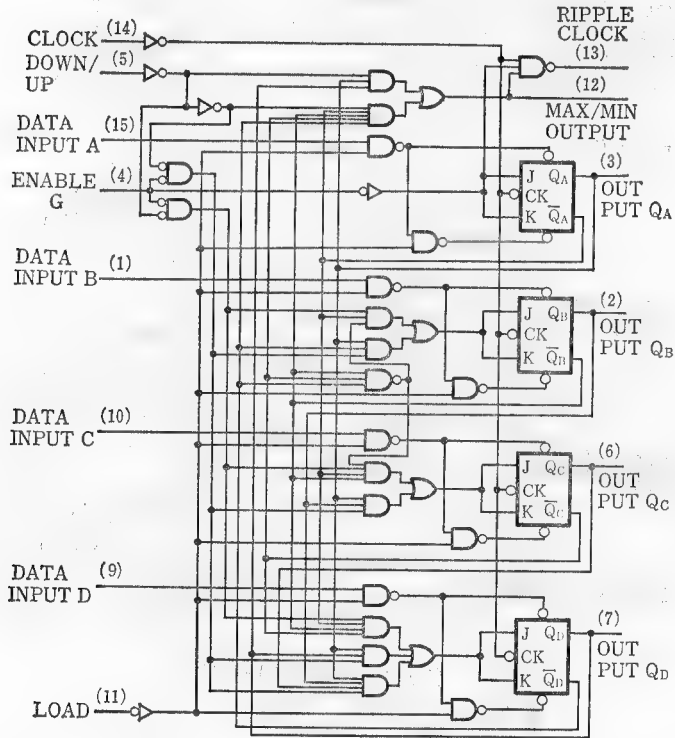
ロードに関しては168, 169がクロック同期，190〜193は非同期である．図8.5には190, 191の内部回路を示す．

（a） 74190 (N, LS)-10進-

図 8.5(a)　同期式カウンタ MSI 内部回路

8.3　シフトレジスタ MSI

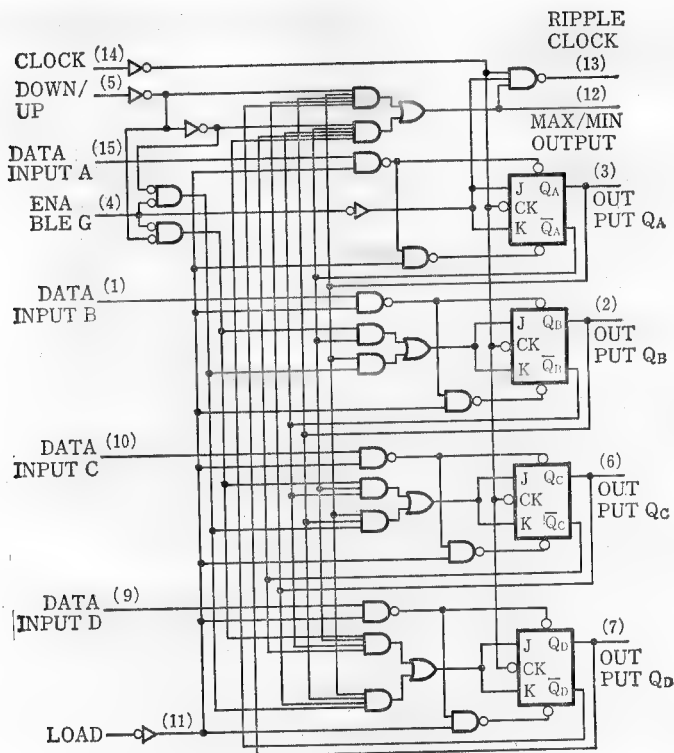
シフトレジスタはFFを縦続接続し，データをシフトしていく機能を持たせたものである．これはデータの直列・並列の変換に使える．

（b） 74191 （N, LS）-16進-

図 8.5（b） 同期式カウンタ MSI 内部回路

図8.6には直列データを並列データに変換することのできるシフトレジスタ74164(N, LS, HC, …)のピン配置および内部回路を示す．A, B端子から順次クロックに同期して入力されたデータは出力端子Q_A～Q_Hから並列的に出力される．これとは逆に，並列データを直列データに変換できるシフトレジスタ74165(N, LS, HC, …)を図8.7に示す．並列にデータを入力するためには，シフト／ロード端子を“L”にする．また，クロック禁止端子を“H”にしたのちシフト／ロード端子を“H”にするとシフトレジスタはホールド状態となり，たとえクロックが入力されてもシフトは起らない．シフト／ロード端子が“H”でクロック

(a) ピン配置

(b) 内部回路

図8.6 8ビットシフトレジスタ74164(N, LS, HC, …)

（a）ピン配置

（b）内部回路

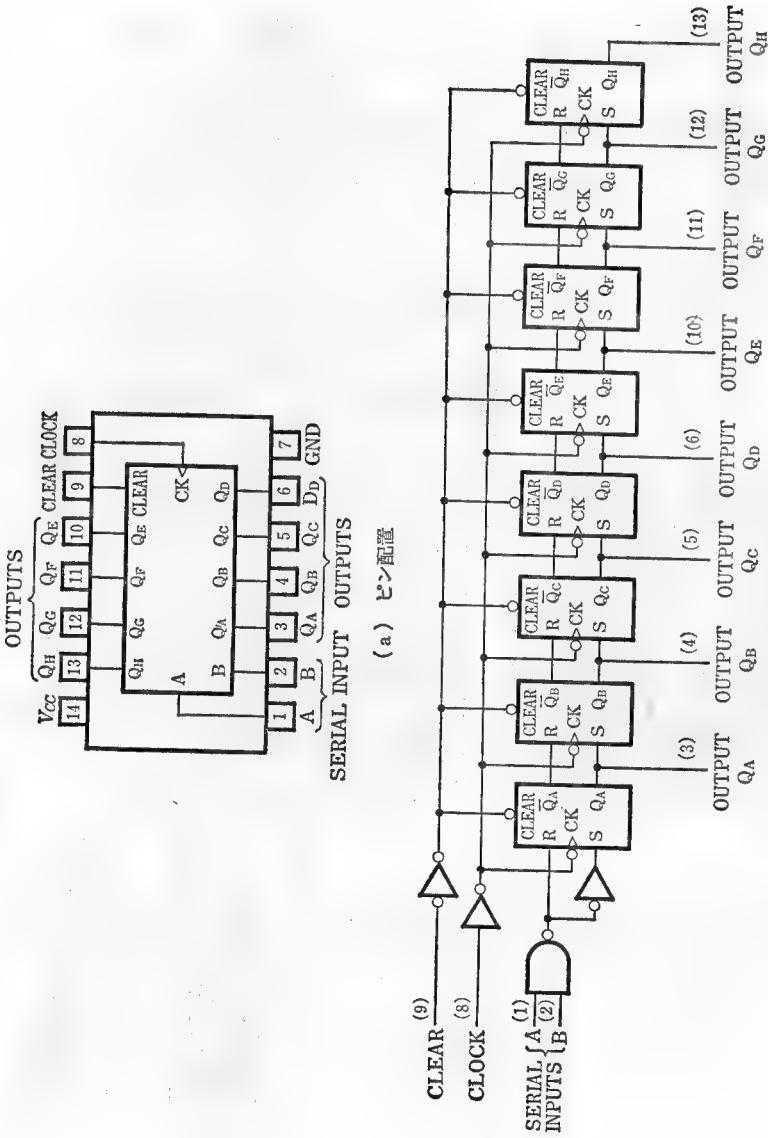
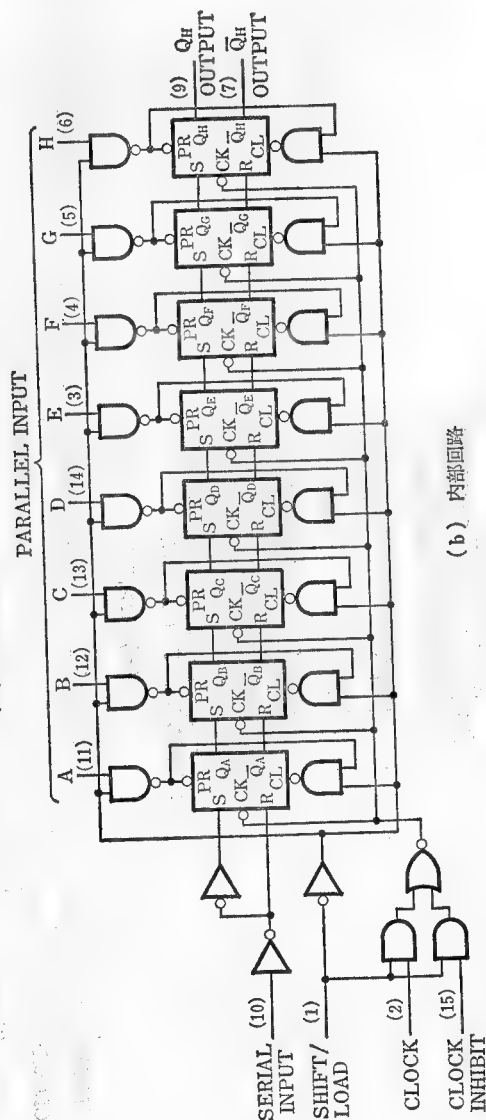
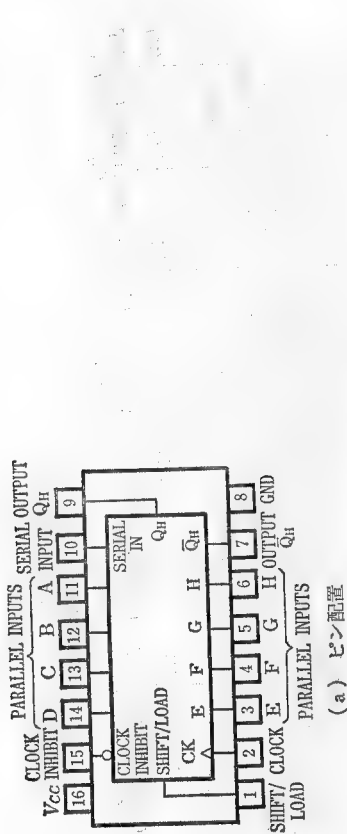



図 8.7　8ビットシフトレジスタ 74165(N, LS, HC, …)

禁止端子が“L”の場合にはクロックパルスによりデータのシフトが起こる.

8.4　可逆シフトレジスタ

通常のシフトレジスタは一方向のみにデータをシフトする機能を持っているが，両方向にデータをシフトできるシフトレジスタも構成できる．これを**可逆シフトレジスタ**という.

可逆シフトレジスタの構成原理図を図8.8に示す．すなわち，各段のFFの入力を左のFFの出力から入れる(右シフト)か，右のFFの出力から入れる(左シフト)かの選択をスイッチで行えばよい．実際の3ビット可逆シフトレジスタの構成例を図8.9に示す．右／左シフト切換え入力が0のとき右シフト，1のとき左シフトとなる.

図8.10には4ビット可逆シフトレジスタMSI 74194(N, LS, HC, …)のピン配置および内部回路を示す．また，動作表を表8.3に示す．左右シフトの切換

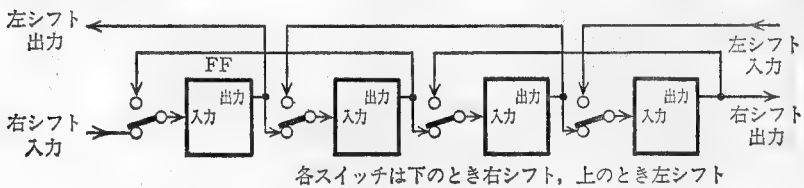



図 8.8　可逆シフトレジスタ原理

図 8.9　可逆シフトレジスタ

（a）ピン配置　74194(N, LS, HC, …)

（b）内部回路　74194(LS, HC, S)

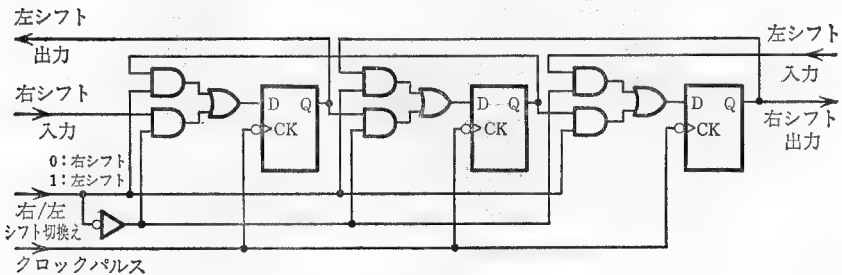
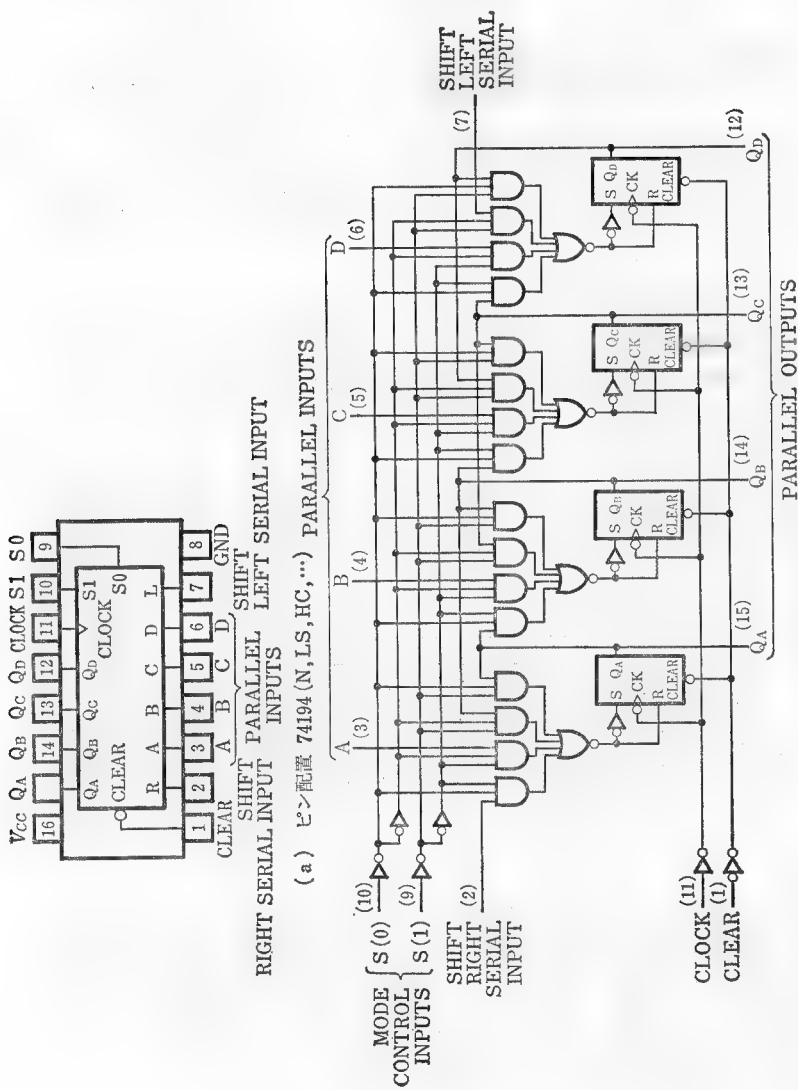



図 8.10　4 ビット可逆シフトレジスタ 74194

えはmode controlにより行われる．このほか，可逆シフトレジスタMSIには8ビットの74198や，左右シフトのクロック入力端子が別個になっている7495(N, LS, HC, …)などがある．ただし，7495を可逆シフトレジスタとして利用するためには外部接続を必要とする．

表 8.3　4ビット可逆シフトレジスタ74194 (N, LS, HC)の動作表

入力					動作
Clear	Mode control		CK		
	S1	S0	N	LS,HC	
H	L	H	↑	↑	右シフト
H	H	L	↑	↑	左シフト
H	H	H	↑	↑	ロード
H	L	L	×	↑	ホールド*
L	×	×	×	×	クリア

*Nタイプはクロック L の期間中S1＝S0＝Lにすると1ビットシフトしてホールドする．クロックHの期間中ならそのままホールド．

8.5 リングカウンタ

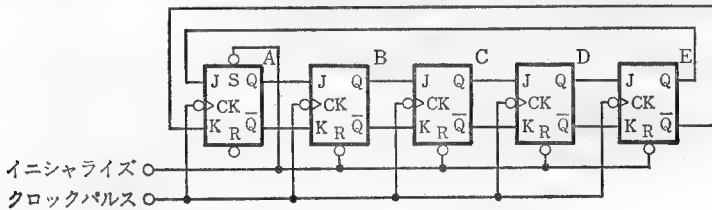
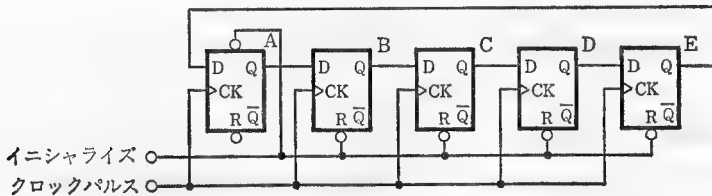
リングカウンタとは，シフトレジスタのシリアル出力をシリアル入力端子に接続，すなわち，円環状に接続し，かつ“1”のパルスが1個だけリング内をぐるぐる回るようにしたものである．リングカウンタの構成例を図8.11に示す．また，その真理値表を表8.4に示す．このカウンタはパルスのカウントよりも

(a) JK-FFを用いた場合

(b) D-FFを用いた場合

図 8.11　5進リングカウンタ

むしろ，そのパルスシフト機能を利用して，図8.12のように多数のデータを選択多重化して1本の線にまとめるマルチプレクサの制御信号などに利用されることが多い．

表 8.4　5進リングカウンタの真理値表

	A	B	C	D	E
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1

なお，上のような構造のリングカウンタでは1個のパルスがリング内をぐるぐる回る構造であるため，途中で誤りが起ってパルスが複数個になったり，消滅したりするとイニシャライズするまでもとに戻らない．そこで，そのような異常状態になると自動的に正常な状態に戻す機能を持った**自己修正形リングカウンタ**もある．また，リングカウンタのFFの一部の入出力線をねじって通常のリングカウンタと異なるパターンを出力するようにした**ツイストリングカウンタ**もある．

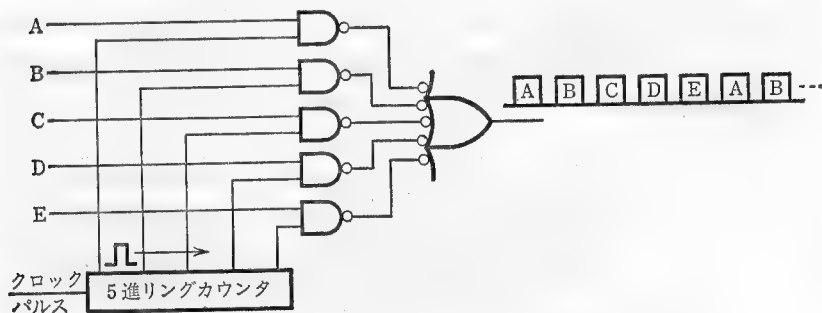



図 8.12　リングカウンタによるマルチプレクサ制御

8.6　レートマルチプライヤ MSI

このMSIの機能概念図を図8.13に示す．たとえば，左から64発のクロックパルスが入るとする．このうち指定された個数のパルスが右から出力される．これは，入力周波数がM/64に分周されたとみることができる．また，入力周波数をf_iとしたとき，f_i/64の周波数がM倍されたと考えることもできる．こ

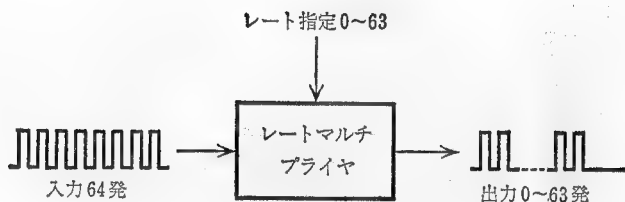



図 8.13　レートマルチプライヤ機能の概念

の意味で，**レートマルチプライヤ**と呼ばれるのである．実際にはレートマルチプライヤの内部では入力パルスを間引くことによりそのような機能を実現しているので，たとえ入力パルスを等間隔で入れたとしても出力パルスは一般に不等間隔となる．

図 8.14　レートマルチプライヤ7497のピン配置

レートマルチプライヤMSIの例として7497をとりあげ，そのピン配置，内部等価回路，機能表をそれぞれ図8.14，図8.15，および表8.5に示す．

図8.15において，図の上半分はカウンタであり，FFによりクロックパルスが順次，分周されていく．Q_Aはクロックパルスが1/2分周されたものである．したがって，レート入力Fがある(“H”)とき，ANDゲートFはクロックパルスの半分を通す．ANDゲートEは$\overline{Q}_A$と$\overline{Q}_B$に接続されているので，ゲート入力Eがあるとき，ゲートFが通さなかったクロックパルスのうち，また半分を通す．以下，各ゲートは残り半分ずつの個数のパルスの出力の可否を決定する．このように出力パルス数Mは入力A～Fに応じてつぎのように決まる．

$$出力パルス数M=2^5\cdot F+2^4\cdot E+2^3\cdot D+2^2\cdot C+2^1\cdot B+2^0\cdot A$$

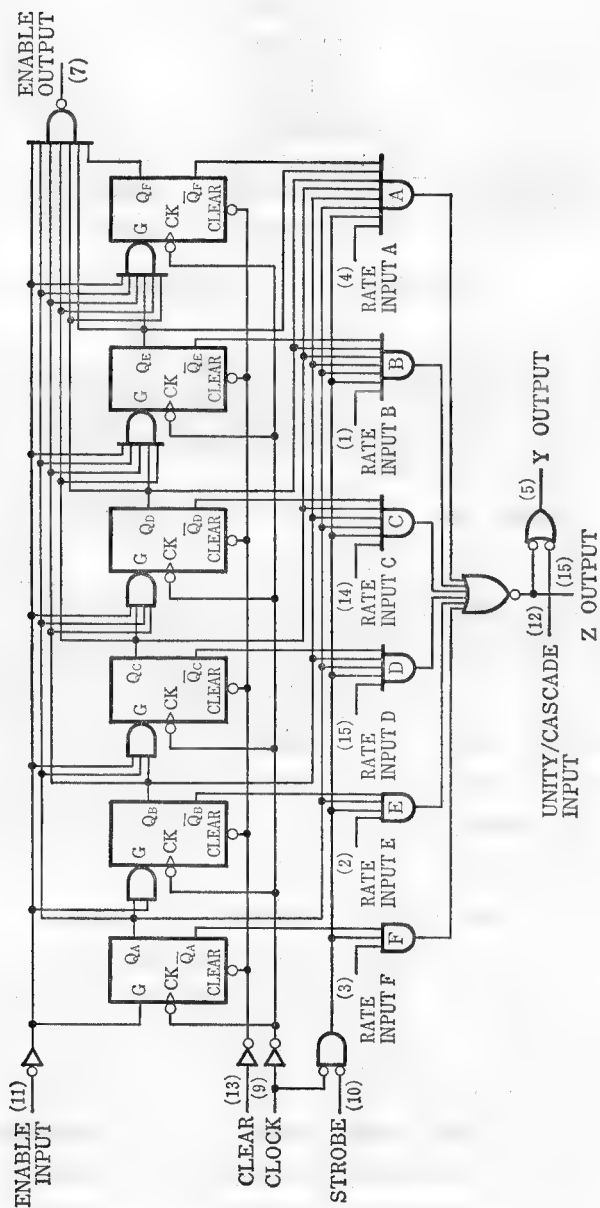



図 8.15　レートマルチプライヤ 7494 の内部回路

表 8.5　7497の機能表

入　力											出　力		
CLEAR	ENABLE	STROBE	BINARY RATE						クロックパルス数	UNITY/CASCADE	論理レベル／パルス数		
			F	E	D	C	B	A			Y	Z	ENABLE
H	×	H	×	×	×	×	×	×	×	H	L	H	H
L	L	L	L	L	L	L	L	L	64	H	L	H	L
L	L	L	L	L	L	L	L	H	64	H	1	1	L
L	L	L	L	L	L	L	H	L	64	H	2	2	L
L	L	L	L	L	L	H	L	L	64	H	4	4	L
L	L	L	L	L	H	L	L	L	64	H	8	8	L
L	L	L	L	H	L	L	L	L	64	H	16	16	L
L	L	L	H	L	L	L	L	L	64	H	32	32	L
L	L	L	H	H	H	H	H	H	64	H	63	63	L
L	L	L	H	H	H	H	H	H	64	L	H	63	L
L	L	L	H	L	L	H	L	L	64	H	36	36	L

演習問題

[8.1] つぎの語句を説明せよ.

（1） シフトレジスタ

（2） リングカウンタ

[8.2] $\overline{\mathrm{CLR}}$パルスを1個以上入れた後，Aにパルスをちょうど5個，Bにパルスをちょうど7個この順に入れ，さらにCから入力“1”を入れると，その間出力が“1”になる回路を，BCDカウンタ74160(図8.16)およびゲート類を用いて構成せよ．明示していない端子の措置は示さなくてもよい．

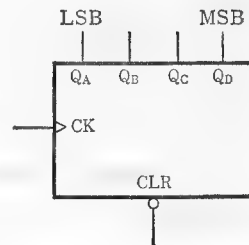



図 8.16　BCDカウンタ 74160(N, LS, HC, …)

9　演算回路

9.1　加　算　器

最も基本的な演算回路であり，減算器や乗算器の基本ともなる．

9.1.1　半加算器

加算器の基本となる回路が**半加算器(ハーフアダー; half adder; HA)**である．これは，1ビット同士の加算を行うことのできるものである．1ビット同士の加算では1+1となったとき，桁上げが起る．1ビット2進数XとYの和をS，桁上げをCとしたときの真理値表を表9.1に示す．SはExclusive ORで構成できる．また，Cは入力XとYがともに1のとき，1を出せばよい．したがって，半加算器は図9.1の回路で実現できる．

表 9.1　半加算器の真理値表

2 進 数		和	桁上げ
X	Y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

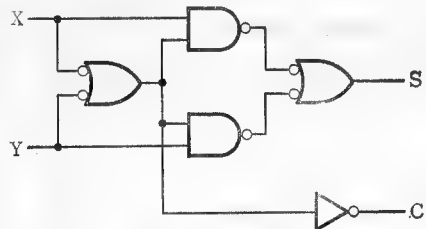



図 9.1　半加算器

9.1.2　全加算器

半加算器は下の桁からの桁上げ(キャリー)を考慮しない構成になっている．図9.2および9.3に示

すように，多ビット同士の加算では下位の桁からの桁上げが生じる．この下位からの桁上げも処理できる1ビット分の加算器が**全加算器**(**フルアダー**；full adder；FA)である．図9.3においては$k=0, 1, \cdots, n-1$について

図 9.2　4ビット同士の加算例

$$S_k=C_{k-1}\oplus X_k\oplus Y_k \quad (9.1)$$

$$C_k=C_{k-1}X_k+X_kY_k+Y_kC_{k-1} \quad (9.2)$$

$$\begin{array}{rrrrr} & C_{n-1}C_{n-2}\cdots C_2 & C_1 & C_0 & \\ & X_{n-1}\cdots\cdots & X_2 & X_1 & X_0 \\ +) & Y_{n-1}\cdots\cdots & Y_2 & Y_1 & Y_0 \\ \hline C_{n-1} & S_{n-1}\cdots\cdots & S_2 & S_1 & S_0 \end{array}$$

図 9.3　2進数の加算

である．ただし，$C_{-1}=0$とする．また，$\oplus$はmod 2の加算で，Exclusive OR，またはHAのS出力を使って実現できる．これらより，全加算器は図9.4のように，半加算器2個を使って実現することができる．全加算器の真理値表を表9.2に示す．

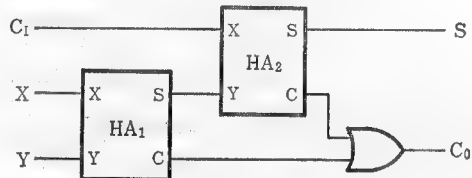



図 9.4　全加算器

表 9.2　フルアダー真理値表

C_I	X	Y	S	C_0
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

9.1.3　並列加算器

複数個のFAを並べることにより，多ビット2進数同士の加算を並列的に行うのが，並列加算回路である．これは，図9.5のような回路で実現できる．ただし，キャリーに関しては直列的に加算されるため，桁上げが続くとき，時間遅れが生じる．そのため，このような形式のキャリーを**リップルキャリー**という．図9.5において，LSBにはキャリー入力がないので，入力は0としておく．または，LSBのみFAの代りにHAを用いてもよい．並列加算を行うためのMSIとして2ビットの7482，4ビットの7483

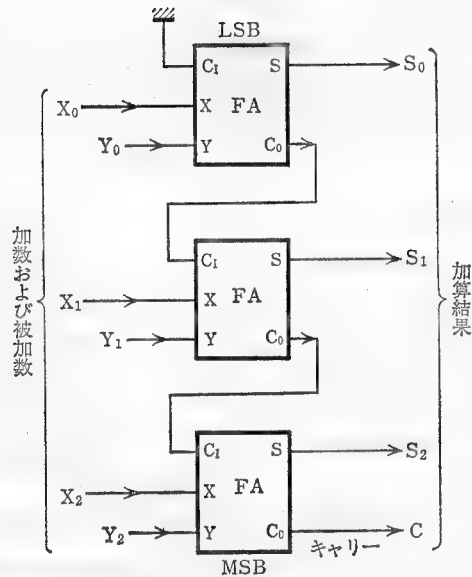



図 9.5　並列加算回路

(N, LS), 74283(N, LS, HC)がある．図9.6および9.7にはこれらのMSIのピン配置と内部回路を示す．

9.1.4　キャリールックアヘッド方式

キャリーに関しても並列に演算を行い，時間遅れをなくした方式は**キャリールックアヘッド(キャリー先見)**方式と呼ばれる．この方式について以下に述べる．

加数をX，被加数をYとするとき，nビット目の加算で発生するキャリーC_nは

$$C_n = X_n Y_n + (X_n + Y_n) C_{n-1} = G_n + P_n C_{n-1} \qquad (9.3)$$

ここに

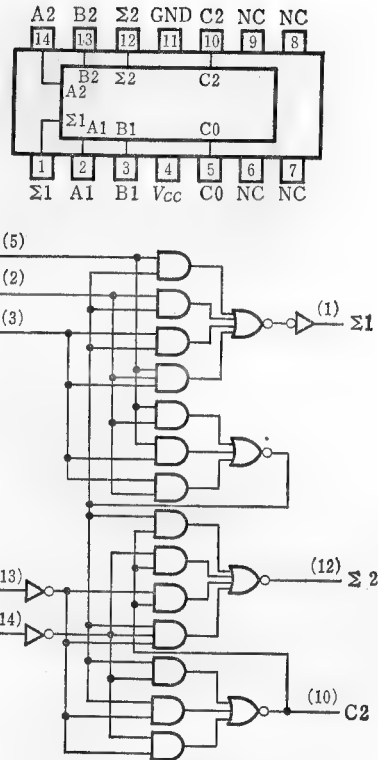
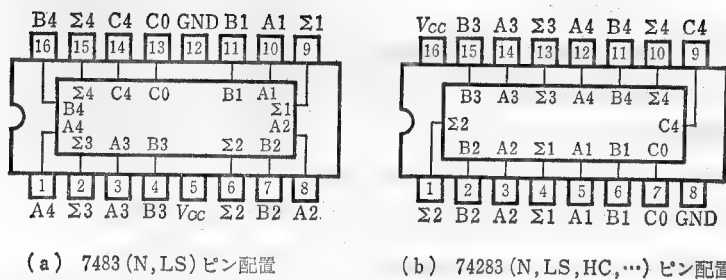
$$\left.\begin{aligned} G_n &= A_nB_n \\ P_n &= A_n + B_n \end{aligned}\right\} \quad (9.4)$$

で与えられる．G_nはnビット目で発生するキャリーを表しており，**キャリー発生関数**(carry generate function)と呼ばれる．また，P_nは，下位の桁からのキャリーを上位の桁へ伝える役割をする関数で，**キャリー伝播関数**(carry propagate function)と呼ばれる．式(9.3)を展開すると

$$\begin{aligned} C_n = G_n & \\ &+P_nG_{n-1} \\ &+P_nP_{n-1}G_{n-2} \\ &+P_nP_{n-1}P_{n-2}G_{n-3} \\ &+\cdots \\ &+P_nP_{n-1}P_{n-2}\cdots P_1G_0 \\ &+P_nP_{n-1}P_{n-2}\cdots P_1P_0C_{-1} \end{aligned} \quad (9.5)$$

図 9.6　2ビットフルアダー7482

この式において，各P, Gは伝播時間遅れなくつくれるから，C_nも伝播時間遅れなくつくれることがわかる．ただし，実際には上の式をこのまま回路実現することは，とくにnが大きくなったとき，困難になる．そのようなときには全ビットをブロックに分割し，各ブロック内および各ブロック間でキャリーの先見を行う．

キャリールックアヘッド回路MSI 74182を図9.8に示す．最上位キャリーは図9.9のようにして求める．この回路は，図9.10のように使用する．

(a) 7483 (N, LS) ピン配置

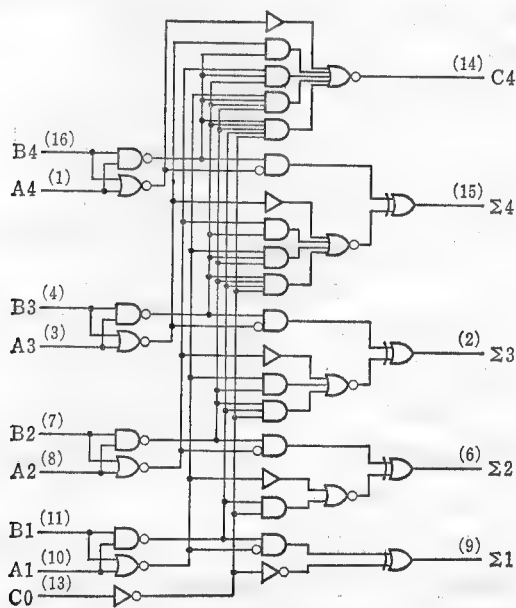
(b) 74283 (N, LS, HC, …) ピン配置

(c) 7483A (LS83A) 内部回路．74283等はピン番号のみ異なる．

図 9.7　4ビットフルアダー

9.1.5　直列加算回路

直列加算を行うと時間はかかるが，とくに大きなビット数のデータに対しても少ない構成要素で実現できる．図9.11に回路を示す．

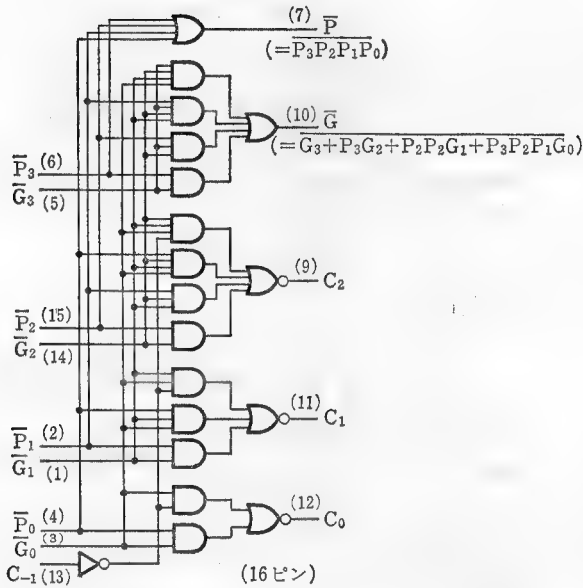



図 9.8　キャリールックアヘッド回路 MSI 74182 (N, LS, HC, …)

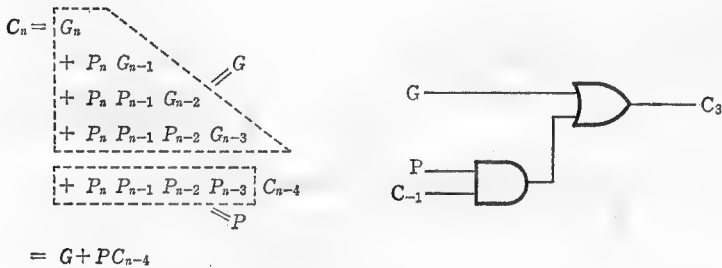
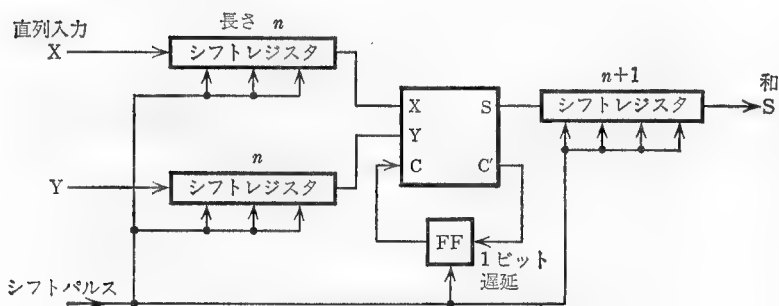
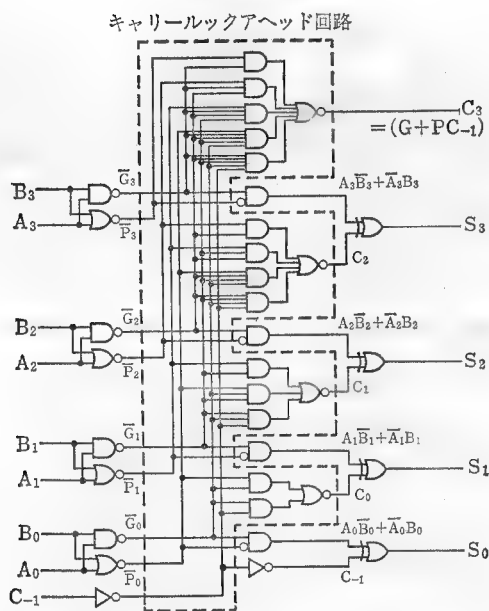



図 9.9　74182 における最上位キャリーの計算

9.2 減算回路

減算回路には直接減算を行う方法と補数を用いて行う方法がある．小規模な減算専用の回路では直接法が，また，大規模・複雑な回路では加減算を統一的

図 9.10　4ビットキャリールックアヘッド加算回路

図 9.11　直列加算回路

に扱うことのできる補数を用いる方法が使われることが多い．

9.2.1　半減算器

1ビットの数同士の減算を考えてみよう．この真理値表を表9.3に示す．こ

れからわかるように，2数の差DはExclusive OR回路で求めることができる．ただし，これ以外に**桁借り**B(borrow)が必要である．これらを実現する回路を図9.12に示す．これを**半減算器(ハーフサブトラクタ**；half subtractor；**HS**)と呼ぶ．

表 9.3　半減算器の真理値表

2進数 X	2進数 Y	差 D=X−Y	桁借り B
0	0	0	0
1	0	1	0
0	1	1	1
1	1	0	0

点線内はExclusive ORと同じ

9.2.2 全減算器

多ビットの数の間の減算を考える．各対応するビットの間では，1ビット数同士の減算が行われるが，このとき，下位の桁からの桁借りが入ってくる．この桁借り入力Bも含めた場合の1ビット数減算の真理値表を表9.4に示す．また，これを実現する全減算器(**フルサブトラクタ**；full subtractor；**FS**)回路を図9.13に示す．

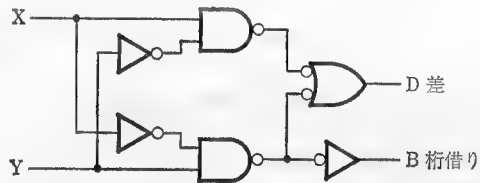



図 9.12　半減算器

表 9.4　全減算の真理値表

入力 X	入力 Y	入力 B	出力 D	出力 B_0
0	0	0	0	0
1	0	0	1	0
0	1	0	1	1
1	1	0	0	0
0	0	1	1	1
1	0	1	0	0
0	1	1	0	1
1	1	1	1	1

9.2.3 並列減算器

一般の減算，たとえば44−25を2進数演算で実現すると図9.14のようになる．このような2つの2進数の減算は加算の場合と同じく，図9.15のような並

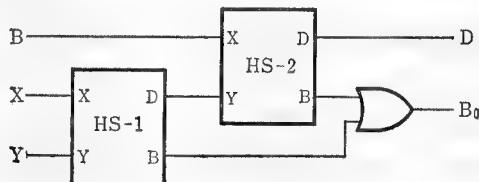



図 9.13　全減算器

一般形

	X	X_3	X_2	X_1	X_0
$-)$	Y	Y_3	Y_2	Y_1	Y_0
		X_3-Y_3	X_2-Y_2	X_1-Y_1	X_0-Y_0
$-)$		B_2	B_1	B_0	
		$X_3-Y_3-B_2$	$X_2-Y_2-B_1$	$X_1-Y_1-B_0$	X_0-Y_0

被減数	X
減数	Y
中間差	D'
ボロー	B
答	D

例

	44	1	0	1	1	0	0	X
$-)$	25	0	1	1	0	0	1	Y
		1	1	0	1	0	1	D'
$-)$		1			1	1		B
	19	0	1	0	0	1	1	D

図 9.14　多ビットの2進数同士の減算（矢印はボローの出た原因を示す）

列減算回路や，直列減算回路で実現できる．しかしながら，2つの数の減算結果が負になる場合まで考慮すると，負数については補数表示を行い，加減算をすべて加算のみで統一的に行う以下の方式が便利である．

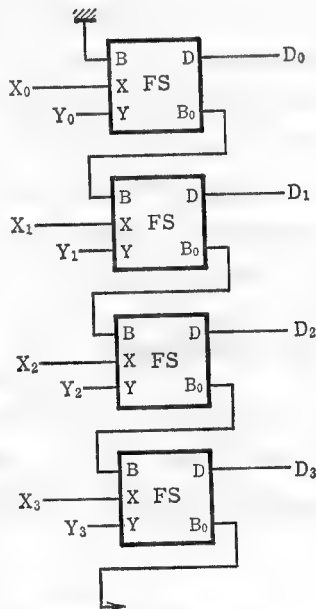
図 9.15　並列減算回路

9.2.4　補数による減算

基準となる数Bからある数yを引いたものを，その数の**補数**という．補数による減算とは，減数yをその補数$(B-y)$で置き換え，被減数xとの加算を行うことにより，減算を行うものである．すなわち，いま$x-y$を求めたいとする．

（i）　$-y$を補数表示$(B-y)$にしておく．

（ii）　$x+(B-y)$を求める．

（iii）　この結果，$B+(x-y)$が得られるので，これからBを引けば減算結果$x-y$が得られるわけであるが，$x-y$が正の場合支障なくそれを行うことができる．

（iv）$x-y$が負の場合には，得られた結果は求めたい結果の補数表示となっている．

ここでは，以下のように左の数が右のような**補数表示**に対応していることがわかる．

yを引くという操作すなわち $-y \longleftrightarrow (B-y)$

$x-y \longleftrightarrow B+(x-y)$

したがって，たとえば$B-y$は，yの補数であり，またそれは$-y$の補数表示である．

このようにすると，加算と減算，および正数と負数が統一的に表現できる．ただし，補数を得る操作・回路が必要となり，そのためその操作が簡単な1の補数や2の補数が通常使われる．

以下では，補数に対してもとの数を**原数**と呼ぶ．また，減算の結果，補数の形で求まった答の補数をとってもとの数に戻したものを**真数**と呼ぶ．なお，ある数xの補数の補数は$B-(B-x)=x$より原数に等しい．すなわち，真数は原数に等しい．

9.2.5　1の補数

n桁の2進数に対して**基準数**を2^n-1として補数をとったものが**1の補数**である．たとえば，4ビット2進数の場合，基準数は$(2^4-1)=15_{10}=1111_2$となる．したがって，2進数1101_2に対する1の補数は$1111_2-1101_2=0010_2$となる．これは，原数の1と0を入れ換えただけであることがわかる．すなわち

原数	1 1 0 1
	↓ ↓ ↓ ↓
1の補数	0 0 1 0

一般に，2進数の1の補数は各ビットを反転することにより得られるのである．

9.2.6　2の補数

n桁の2進数に対して基準数を2^nとして補数をとったものである．したが

って，「2の補数」というよりも「2^nの補数」であると覚えたほうが理解しやすい．たとえば，4ビット2進数の場合，基準数は$2^4=16_{10}=10000_2$となる．したがって，2進数1101_2に対する**2の補数**(2^4の補数)は$10000_2-1101_2=0011_2$となる．

1の補数の基準数2^n-1と2の補数の基準数2^nとの差は1である．したがって，それらによる補数の差も1となる．上の例では

$$\begin{array}{rrrrrl} & 0 & 0 & 1 & 0 & \cdots 1\text{の補数} \\ -) & 0 & 0 & 1 & 1 & \cdots 2\text{の補数} \\ \hline & & & & -1 & \end{array}$$

1の補数は1と0を反転することにより求められた．したがって，2の補数は原数の1と0を反転し，それ（1の補数)に1を加えれば求まる．

9.2.7　2の補数による減算

すでに述べたように，被減数をx，減数をy，基準数をBとしたとき，$x-y$の減算は

$$x-y=x+(B-y)-B$$

のようにして行うことができる．すなわち，xにyの補数を加えて最後に基準を引けばよい．以下ではよく使われる2の補数による減算の具体例について述べる．

2の補数を扱う場合，基準数Bは

$$2^n=1\underbrace{000\cdots0}_{n\text{個}}$$

となる．したがって，減算結果が正数の場合，xにyの補数を加えた和から，$n+1$ビット目の1をとればよい．

【例】 $n=4$，$x=13$，$y=6$の場合，図9.16のようになる．

つぎに，減算結果が負になる場合を考えてみよう．

$$Z=x+(B-y)=B+(x-y)$$

としたとき

$$x-y=x+(B-y)-B=Z-B$$

$$\begin{array}{rl} & x=1101_2 \\ -) & y=0110_2 \\ \hline \end{array} \Longrightarrow \begin{array}{rl} & x=1101 \\ +) & y\text{の補数}=1010 \\ \hline & ①\vdots 0111 \end{array}$$

ここが1であるから結果は正数であることがわかる.

取り去る

答 $0111_2=7_{10}$

図 9.16 2の補数による減算(結果が正の場合)

はZの補数に負号をつけたものになっている．別な見方をすれば，xにyの補数を加えたZは減算結果の絶対値の補数をとったものになっている.

すなわち，Zは減算結果$x-y$の補数表示になっている．このような場合の例を図9.17に示す.

【例】 いま$n=4$, $x=6$, $y=13$とする.

$$\begin{array}{rr} & 6 \\ -) & 13 \\ \hline & -7 \end{array} \Longrightarrow \begin{array}{rl} & x=0110 \\ -) & y=1101 \\ \hline \end{array} \Longrightarrow \begin{array}{rl} & x=0110 \\ +) & y\text{の補数}=0011 \\ \hline & z=0\vdots 1001 \end{array}$$

ここが0であるから結果は負数であることがわかる

反転

0110

+) 1

0111

真数に直している

減算結果は−7である

図 9.17 2の補数による減算(結果が負の場合)

上の例では，減算結果が負数の場合，真数を得るために再度補数演算を施した．しかしながら，多くの加減算を統一的に扱うためにはそのつど真数を得るのではなく，負数はできるだけ補数のままにしておいたほうが便利である．そして，必要なときのみ真数に戻すのである．そのために，MSBの1つ上の第$n+1$ビット目が符号桁に対応するように表9.5のような補数表示を用いる．すなわち，絶対値がnビット2進数で表される正または負の数aを

$$b=\begin{cases} a & a\geq 0 \text{のとき} \\ 2^{n+1}+a & a<0 \text{のとき} \end{cases}$$

なるbで表す．別の表現をすれば

$$\begin{aligned} b&=B'+a \\ &=2^{n+1}+a \pmod{2^{n+1}} \end{aligned}$$

表 9.5　符号桁を含めた補数表示(n=3)

10進数表示	絶対値表示	補数表示
+7	+111	0111
+6	+110	0110
+5	+101	0101
+4	+100	0100
+3	+011	0011
+2	+010	0010
+1	+001	0001
0	+000	0000
−1	−001	1111
−2	−010	1110
−3	−011	1101
−4	−100	1100
−5	−101	1011
−6	−110	1010
−7	−111	1001

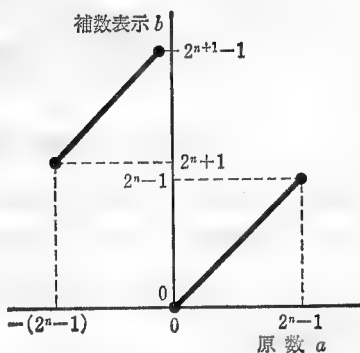



図 9.18　符号桁を含めた 2 の補数表示

ここに，B' は基準数 2^{n+1} である．

この補数表示と原数の関係を図示すれば図 9.18 のようになる．

符号桁を持つ(符号桁も入るよう 1 ビット長くした)補数表示を用いると，加減算とも同じ加算の形式で実行できる．ただし，負数は補数の形で表されることになる．図 9.19 に例を示す．

これらと，符号桁を使わない減算を比べてみると，符号桁を使うほうが符号桁のため 1 ビット長い演算をしなければならないが，演算の形式は両者同じであり，符号桁を使うほうが演算の制御が楽になる．

9.2.8　オーバフローとアンダフロー

演算結果が表示可能範囲を超えると，オーバフローまたはアンダフローが発生し，正確な解が求まらなくなる．

【例】 n=4 のとき，これに符号桁をつけても表示可能範囲は −15～+15 である．結果がこの範囲を越える加減算ではオーバフロー，アンダフローが発生する．

【例 1】 $n=4$, $x=13$, $y=6$(減算結果が正)の場合

```
      x= 0 1 1 0 1                    x= 0 1 1 0 1
 −)   y= 0 0 1 1 0   ⟹     +)  yの補数= 1 1 0 1 0
                                  ①⓪0 1 1 1  (補数表示の解)
                       無視する ──┘ ↓
                                  + 0 1 1 1₂ = +7₂(絶対値表示の解)
```

【例 2】 $n=4$, $x=6$, $y=13$(減算結果が正)の場合

```
      x= 0 0 1 1 0                    x= 0 0 1 1 0
 −)   y= 0 1 1 0 1   ⟹     +)  yの補数= 1 0 0 1 1
                                   ① 1 0 0 1  (補数表示の解)
                                   │  ↓反転
                                   │  0 1 1 0
                                   │ +)    1
                                   └→ −0 1 1 1₂ = −7₁₀  (絶対値表示の解)
```

【例 3】 $n=6$, $x=52$, $y=17$(減算結果が正)の場合

```
          符号桁                     符号桁
            ↓                          ↓
          x=0 1 1 0 1 0 0              0 1 1 0 1 0 0
 −)       y=0 0 1 0 0 0 1   ⟹    +) 1 1 0 1 1 1 1
                                      ①⓪1 0 0 0 1 1
                         mod 2⁷に ──┘ ↓
                         より無視    = + 1 0 0 0 1 1  (mod 2⁷)
                                     = +35₁₀
```

【例 4】 $n=6$, $x=17$, $y=52$ (減算結果が負)の場合

```
      x= 0 0 1 0 0 0 1              0 0 1 0 0 0 1
 −)   y= 0 1 1 0 1 0 0   ⟹    +) 1 0 0 1 1 0 0
                                    ①0 1 1 1 0 1  (解の補数表示)
                                    │  ↓反転
                                    │   1 0 0 0 1 0
                                    │ +)          1
                                    └→ −1 0 0 0 1 1₂
                                       =−35₁₀          (解の絶対値表示)
```

図 9.19 符号桁を持つ補数表示による減算例

```
     10                  0 1 0 1 0
 +)   7      ⟹     +) 0 0 1 1 1
                         1 0 0 0 1
                         └───────┘
                         −15となってしまった(オーバフロー)
```

```
   −10          1⋮0110
−)  +7    ⟹  +) 1⋮1001
              ──────────
              10⋮1111
               +15となってしまった(アンダフロー)
```

オーバフロー，アンダフローに対する対策は，演算のビット長をもう1ビット長くしておくことである．すなわち，つぎのようにすればよい．

```
    10          0⋮01010
+)   7    ⟹  +) 0⋮00111
              ──────────
                0⋮10001
                 +17(オーバフローなし)

   −10          1⋮10110
−) + 7    ⟹  +) 1⋮11001
              ──────────
               11⋮01111
                 −17(アンダフローなし)
```

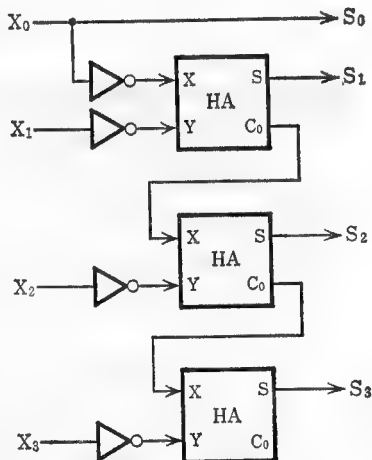
9.2.9　2の補数回路

さきに述べたように，2の補数を得るためには2進数原数の1と0を反転し，1を加えればよかった．したがって，これは図9.20のような回路で実現できる．また，2の補数による減算回路は図9.21のようにして実現できる．

図 9.20　2の補数回路

9.2.10　加減算回路

2の補数を用いて，加減算両方が可能な回路を図9.22に示す．減算の場合，制御信号は1を入れる．このとき，4ビット全加算器には反転した信号が入るが，キャリー入力も入るため，2の補数を入れることになる．

加算のときにはC_0はキャリー出力になっている．減算のときにはC_0が1の

とき減算結果は正，0のときには負で，S_0〜S_3は補数が出力されている．

減算した結果が負で，それを真数に戻す必要があるときには，$S \cdot \bar{C}_0$が1の条件で出力を2の補数回路に通せばよい．

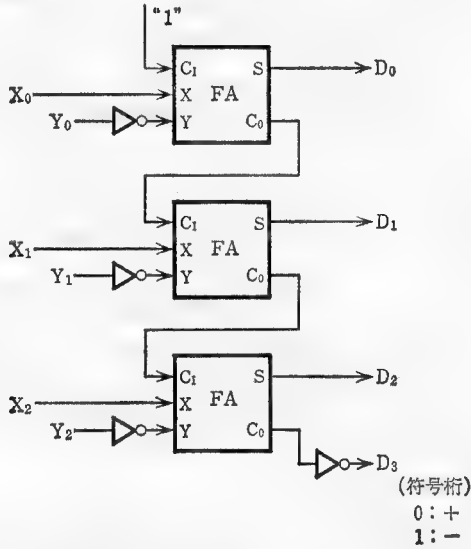



図 9.21　2の補数による減算回路（とくに減算結果が負のとき2の補数で出力される）

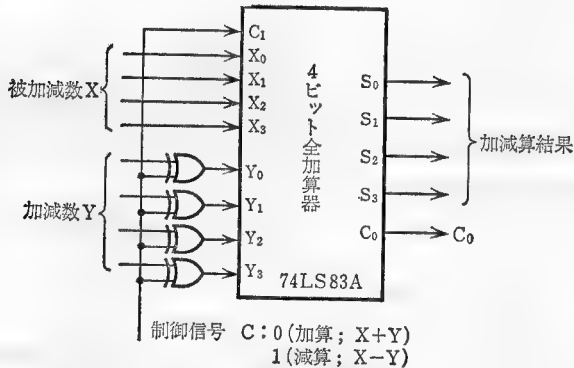



図 9.22　4ビット加減算回路

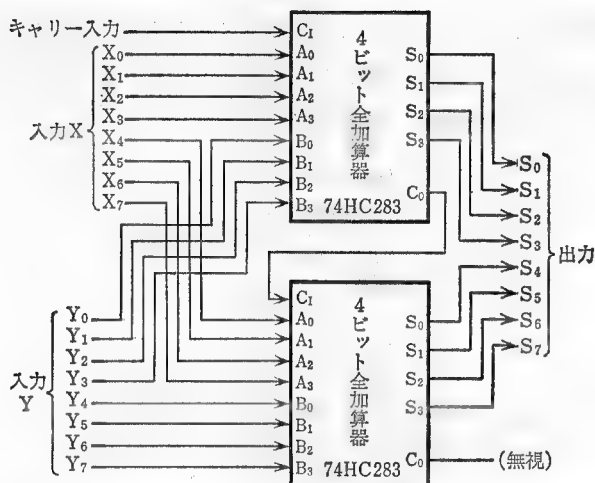



図 9.23　符号桁つき 2 の補数表示を用いる加減算器　MSB(X_7, Y_7, S_7)は符号桁である．入力 X, Y はオーバフロー，アンダフローが生じないように $-(2^6-1)$～$(+2^6-1)$ とし，2の補数表示を用いる．出力は $-(2^7-1)$～$+(2^7-1)$ の符号桁つき 2 の補数表示となる．

符号桁つき 2 の補数表示を用いて 2 数 X, Y の加減算を統一的に行う回路を図 9.23 に示す．入出力とも符号桁は MSB とする．入力 X, Y を -63～$+63$ の範囲にしておけば出力のオーバフロー，アンダフローによる誤りは生じない．出力は -127～$+127$ の範囲の符号桁つき 2 の補数表示となる．

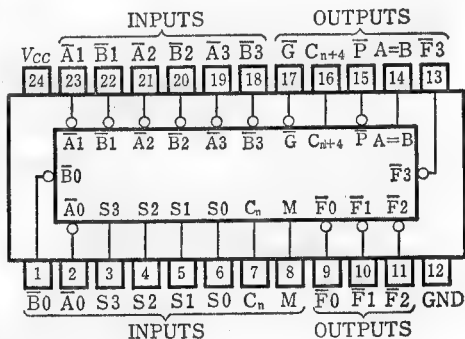



図 9.24　ALU 74181 (N, LS, HC, …) のピン配置

9.2.11　ALU による加減算

コンピュータの CPU (central processing unit) においては加減算や論理演算など各種の基本演算が行われるが，それを受け持つのが **ALU** (arithmetic logic

unit；算術論理演算機構）である．

代表的なALU 74181のピン配置を図9.24に示す．また，端子の意味と機能を表9.6，9.7に示す．たとえば，S3 S2 S1 S0＝LHHL，M＝L，C_n＝Lとすると，AマイナスBを実行できる．図9.25には内部等価回路を示す．加算は内部的にキャリールックアヘッド方式で計算される．74LS181ではFは最大

表 9.6 ALU 74181の端子の意味

記号	機能
A3, A2, A1, A0	入力A
B3, B2, B1, B0	入力B
S3, S2, S1, S0	演算機能選択入力
C_n	キャリー入力
M	モード制御入力
F3, F2, F1, F0	演算出力
A＝B	比較器出力（オープンコレクタ）
P	キャリー伝播出力
C_{n+4}	キャリー出力
G	キャリー発生出力

表 9.7 ALU 74181等の演算機能

機能選択				正論理データに対して		
				M＝H	M＝L（算術演算）	
S3	S2	S1	S0	（論理演算）	C_n＝H（キャリーなし）	C_n＝L（キャリーあり）
L	L	L	L	$F=\bar{A}$	F＝A	F＝A Plus 1
L	L	L	H	$F=\overline{A+B}$	F＝A＋B	F＝(A＋B) Plus 1
L	L	H	L	$F=\bar{A}B$	$F=A+\bar{B}$	$F=(A+\bar{B})$ Plus 1
L	L	H	H	F＝0	F＝Minus 1（2の補数）	F＝Zero
L	H	L	L	$F=\overline{AB}$	F＝A Plus $A\bar{B}$	F＝A Plus $A\bar{B}$ Plus 1
L	H	L	H	$F=\bar{B}$	F＝(A＋B) Plus $A\bar{B}$	F＝(A＋B)Plus $A\bar{B}$ Plus 1
L	H	H	L	$F=A\oplus B$	F＝A Minus B Minus 1	F＝A Minus B
L	H	H	H	$F=A\bar{B}$	$F=A\bar{B}$ Minus 1	$F=A\bar{B}$
H	L	L	L	$F=\bar{A}+B$	F＝A Plus AB	F＝A Plus AB Plus 1
H	L	L	H	$F=\overline{A\oplus B}$	F＝A Plus B	F＝A Plus B Plus 1
H	L	H	L	F＝B	$F=(A+\bar{B})$ Plus AB	$F=(A+\bar{B})$Plus AB Plus 1
H	L	H	H	F＝AB	F＝AB Minus 1	F＝AB
H	H	L	L	F＝1	F＝A Plus A*	F＝A Plus A Plus 1
H	H	L	H	$F=A+\bar{B}$	F＝(A＋B) Plus A	F＝(A＋B) Plus A Plus 1
H	H	H	L	F＝A＋B	$F=(A+\bar{B})$ Plus A	$F=(A+\bar{B})$ Plus A Plus 1
H	H	H	H	F＝A	F＝A Minus 1	F＝A

機能選択				負論理データに対して		
				M=H	M=L(算術演算)	
S3	S2	S1	S0	(論理演算)	C_n=L(キャリーなし)	C_n=H(キャリーあり)
L	L	L	L	$F=\bar{A}$	F=A Milus 1	F=A
L	L	L	H	$F=\overline{AB}$	F=AB Minus 1	F=AB
L	L	H	L	$F=\bar{A}+B$	$F=A\bar{B}$ Minus 1	$F=(A\bar{B})$
L	L	H	H	F=1	F=Minus 1(2の補数)	F=Zero
L	H	L	L	$F=\overline{A+B}$	$F=A$ Plus $(A+\bar{B})$	$F=A$ Plus $(A+\bar{B})$ Plus 1
L	H	L	H	$F=\bar{B}$	F=AB Plus (A+B)	$F=AB$ Plus $(A+\bar{B})$Plus 1
L	H	H	L	$F=\overline{A+\bar{B}}$	F=A Minus B Minus 1	F=A Minus B
L	H	H	H	$F=A+\bar{B}$	$F=A+\bar{B}$	$F=(A+\bar{B})$ Plus 1
H	L	L	L	$F=\bar{A}B$	F=A Plus (A+B)	F=A Plus(A+B) Plus 1
H	L	L	H	F=A+B	F=A Plus B	F=A Plus B Plus 1
H	L	H	L	F=B	$F=A\bar{B}$ Plus (A+B)	$F=A\bar{B}$ Plus(A+B) Plus 1
H	L	H	H	F=A+B	F=A+B	F=(A+B) Plus 1
H	H	L	L	F=0	F=A Plus A*	F=A Plus A Plus 1
H	H	L	H	$F=A\bar{B}$	F=AB Plus A	F=AB Plus A Plus 1
H	H	H	L	F=AB	$F=A\bar{B}$ Plus A	$F=A\bar{B}$ Plus A Plus 1
H	H	H	H	F=A	F=A	F=A Plus 1

＊　左へ1ビットシフトすることを意味する.
注：＋・⊕ーは論理演算，Plus，Minus は算術演算.

38ns，キャリーは最大41ns，で求まる.

9.3　乗算回路

9.3.1　2^n倍および$1/2^n$倍回路

2進数の2^n倍および$1/2^n$倍はシフトレジスタ等によるシフト操作のみで実行できる．2^n倍は左へnビットシフトし，$1/2^n$倍は右へnビットシフトすればよいのである．図9.26に例を示す．2^n倍のときにはオーバフローが起こらないようにしなければならない．そのため図9.26(a)では出力のビット長を長くしてある．ただし，ビット長が長くなっても，シフトに伴って最下位ビットには0が挿入されるだけである．また，$1/2^n$倍のときには右シフトを受けるため，下位ビットは切り捨てられ，上位ビットには0が挿入される．

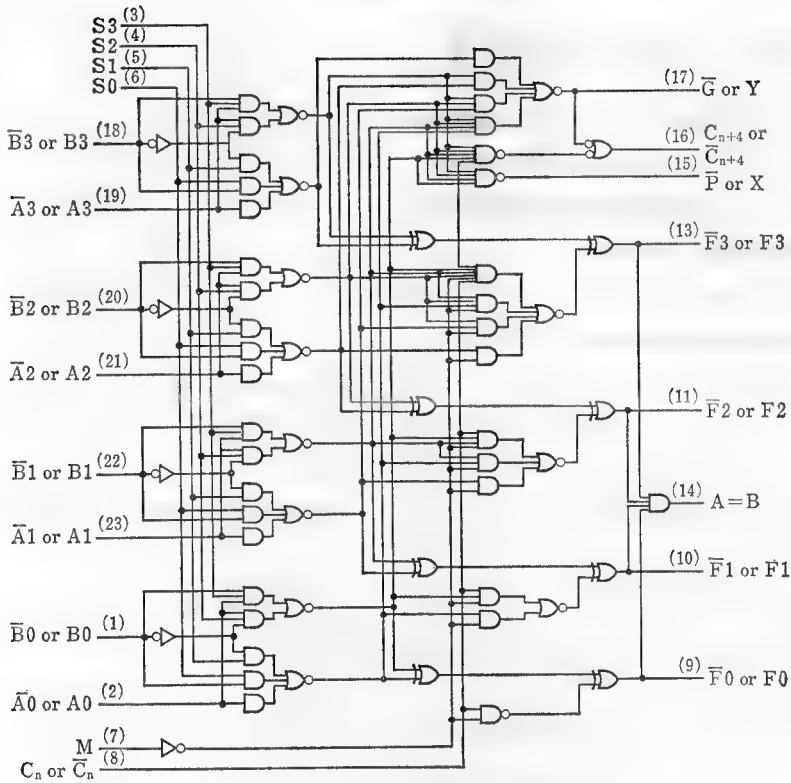



図 9.25 ALU 74181 内部等価回路

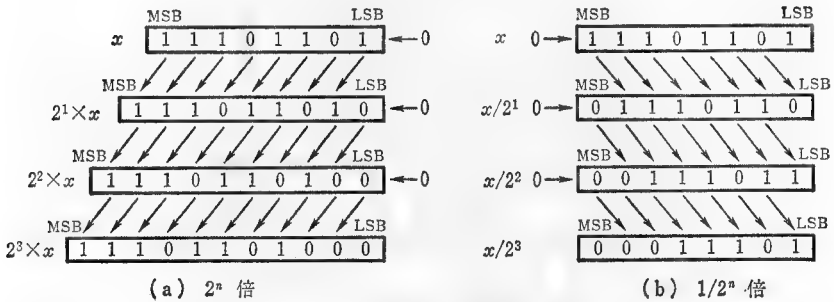



図 9.26 2進数の2^n倍および$1/2^n$倍

9.3.2.　組合せ回路方式乗算回路

2進数同士の乗算は10進数の場合と同じく，図9.27のように実行できる．また，これを実現する**組合せ回路方式乗算回路**を図9.28に示す．

```
  14            1110
×) 13   ⟹   ×) 1101
               1110
              0000
             1110
            1110
          10110110 =182₁₀
```

図 9.27　2進数の乗算

組合せ回路方式により4ビット2進数同士の乗算を行うMSIとして74284および74285がある．74284は上位4桁，74285は下位4桁を計算・出力する．図9.29にこれらのピン配置および使用例を示す．なお各出力はオープンコレクタとなっている．

さらに長い桁同士の乗算や，$\sum_i x_i y_i$ のような**積和計算**を行うCMOS LSIの

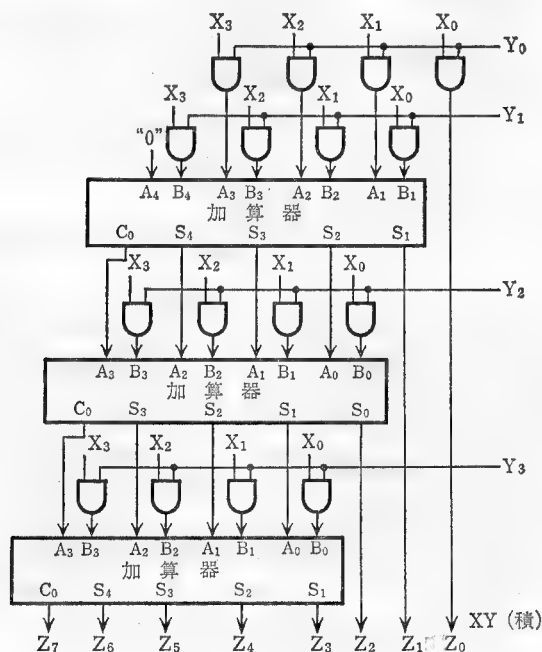



図 9.28　組合せ回路方式の4ビット×4ビット乗算回路

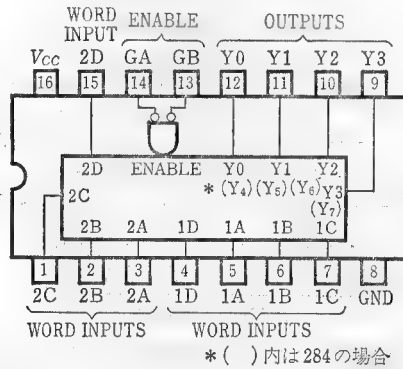
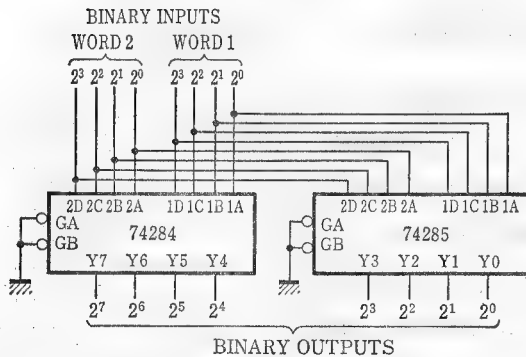
例を図9.30に示す．8ビット×8ビットの積和計算を行うため，出力レジスタは19ビットになっている．**プリロード**機能により，出力ライン上の19ビットデータを積和演算の初期値としてとり込むこともできる．累積制御は乗算だけか，乗算した結果を出力レジスタに足し込むか(加減制御により減算も可)の制御を行う．データ形式制御は2の補数表示か，無符号絶対値表示かの選択を行う．丸め制御はP7に1を加えることにより行う(P8以上が有効)．このほか，16ビット×16ビット乗算器や，32ビット×32ビット**浮動小数点乗算器** LSIなども市販されている．また，高速積和演算器にマイクロプロセッサを組

（a） 74285/284のピン配置

（b） 4×4ビットの並列乗算

図 9.29 乗算MSI 74285/284

み合わせた**シグナルプロセッサ**もある．

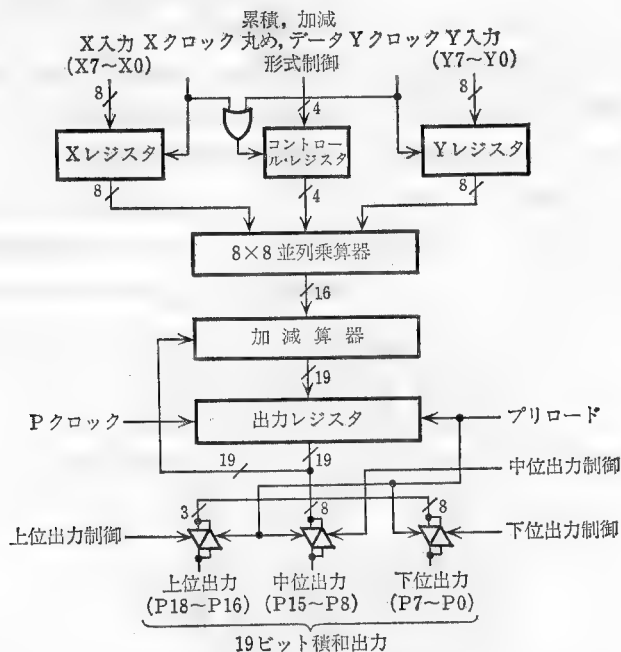



図 9.30　8ビット×8ビットCMOS積和LSI(日本プレシジョンサーキッツ社 SM5810．電源+5V，入出力はTTLコンパチブル．乗算時間45ns)

9.4　除算回路

2進数の除算も10進数と同じように行える．ただし2進数の場合は0と1しかないので，被除数から除数が引けるか，引けないかの2種類しかない．引けた場合には商に1を立て，引けなかった場合には0を立てる．これを順次各桁について行えばよい(図9.31)．

```
                101110 ……商
除数……0101 ) 11101001 ……被除数
              101
               1001
               0101
                1000
                0101
                 0110
                 0101
                  0011 ……余り
```

図 9.31　2進数の除算

この方法には，引けなかった場合，すなわち結果が負になった場合，(a)引いてしまった値を加えたり，別に残しておいた値で引く前の値を回復する方法と，(b)引いた分はそのままにしておいてつぎの桁位置での引算のときに引いてしまった値を加えることによりその精算をする非回復法がある．なお，(b)

(a) 回復法による除算

(b) 非回復法による除算

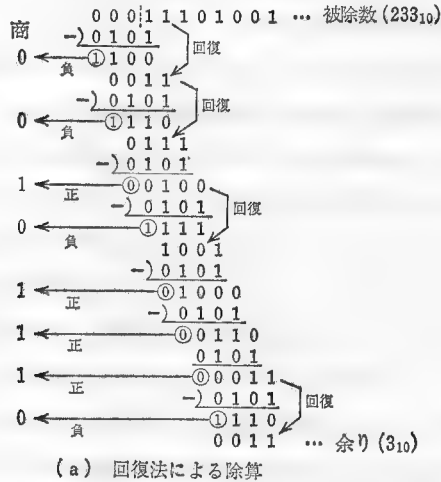



図 9.32 2進数除算の実現法

の次桁位置での精算はつぎのようにして行う．すなわち被除数をx，除数をaとして，xのまま残しておくべきところ$x-a$としてしまったとき，これらは次桁位置では1ビットシフトして考えてそれぞれ①$2x$および②$2(x-a)$となる．この位置で，$2x$からaが引けるかどうかみる($2x-a$をつくる)ことは後者②では$2(x-a)+a$という加算を行うことに相当する．これら(a)(b)の方式に従う除算の例を図9.32に示す．

このうち，(a)の回復法に従うnビット÷mビット除算回路の構成法を図9.33に示す．この回路では最初，被除数が$n+m-1$ビットAレジスタの下位nビット(A_L)に，また除数がmビットDレジスタにセットされる．Aレジスタの上位mビット($=A_U$)からDレジスタの内容を引いたものが正(減算回路出力のMSBが0)なら商として1を立て，それをAレジスタのLSB入力へセットするとともに，減算結果をA_Uに入れる．もしそれが負なら，商として0を立て，それをAレジスタのLSB入力へセットするが，A_Uの内容は変えない．しかる後，Aレジスタの内容を左へ1ビットシフトする．このような操作をn回繰り返せば，商がAレジスタの下位nビット(A_Lレジスタ)に求まり，余りがA_Uレジスタに求まる．実際には，このような減算の実行タイミングやシフトのタイミングとその回数を制御する制御回路が必要となる．この減算回路にお

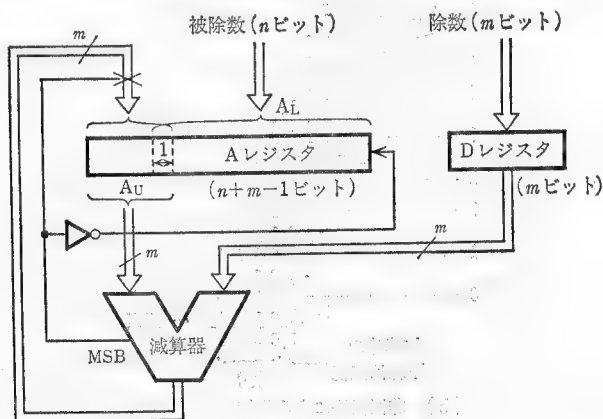



図 9.33　回復法による除算回路

けるAレジスタを中心とするデータの流れを図9.34に示す．

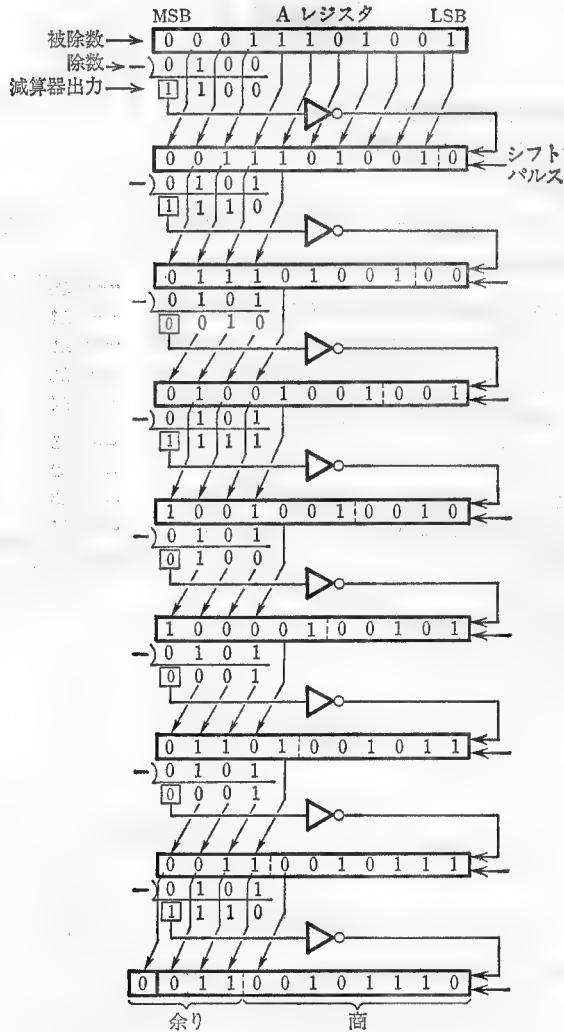



図 9.34 除算(233÷5)のためのレジスタと減算器におけるデータの動き

演習問題

［9.1］ つぎの語句を説明せよ．

（1） ハーフアダー，フルアダー

（2） 2の補数

［9.2］ 正の8ビット2進数（0～2^8-1）同士の乗算結果は何ビットで表現されるか．また，除算ではどうか．

［9.3］ 4ビット2進数（MSBを符号桁とし，負数は2の補数表示されている）同士の加算器を4ビットフルアダー7483（複数個使用可）およびゲート類を用いて構成せよ．結果は符号桁，および絶対値を出力せよ．ただし，オーバフローは生じないものとする．

［9.4］ 正の2ビット2進数同士の乗算回路を，4ビットフルアダー7483，およびゲート類を用いて構成せよ．

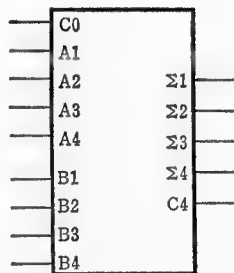



図 9.35　4ビットフルアダー7483

10 記憶素子

IC記憶素子としては種々のものが開発されている．それらを分類すると図10.1のようになる．

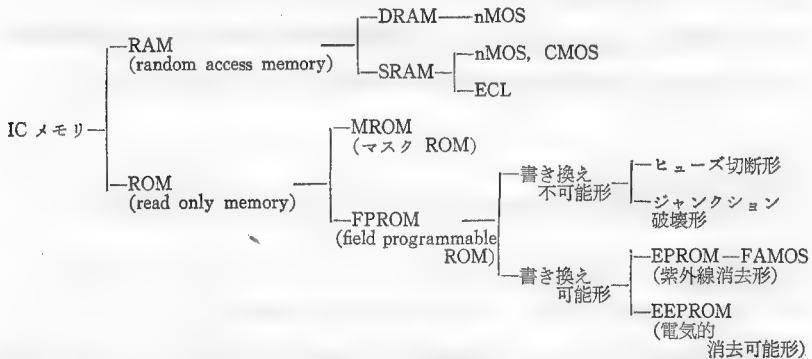



図 10.1　IC記憶素子

10.1　RAMとROM

RAM(random access memory)は代表的なメモリ素子である．正確にはリードライトメモリというべきものであるが，慣習的に**ランダムアクセスメモリ**といわれる．アドレスを指定することにより，任意の記憶場所へ書き込んだり，その内容を読み取ったりすることが自由にできる．ただし，電源を切るとその内容は消失(蒸発)する．

RAMには**DRAM**(dynamic RAM)と**SRAM**(static RAM)がある．DRAM

はSRAMよりも低速でかつ記憶を保持するための特別な操作(リフレッシュ)を必要とするが，素子当りの容量の点で優れている(4倍)ため，コンピュータの主記憶などでよく使われる．一方，SRAMは高速でリフレッシュ操作を必要とせず使いやすいので，小容量メモリや高速でアクセスする必要のある場合によく使われる．

一方，**ROM**(read only memory)もランダムアクセス可能なメモリであるが，電源を切ってもその内容は蒸発しない．読み出しは自由であるが，一般に書き込みと消去(消去可能なものについて)には特殊な操作を必要とする．

このほか，RAMのように自由に読み書きでき，かつ電源を切ってもその内容が消失しない**不揮発性メモリ**もあるが，まだ特殊なものである．

また，**CCD**(charge coupled device)や**磁気バブルメモリ**のように時系列的にデータの読み出し／書き込みができる**シリアルアクセスメモリ**もある．

10.2 DRAM

10.2.1 原　　理

DRAMの基本は，たとえば20 μm^2(1MビットDRAM)というような小さな面積の場所につくられた0.04 pFというような小さな容量のコンデンサに電荷を蓄えることにより記憶動作を行う．容量が非常に小さいため，2 sもすれば蓄積された電荷は消えてしまう．そこで安全をみて，たとえば2 msごとに各コンデンサの内容を読み出し，再度書き込んでやることが必要となってくる．これを**リフレッシュ**という．この操作はユーザが行うが，簡単な制御さえ行えばあとは素子の中で自動的に行うようになっているもの(**擬似SRAM**)もある．

図10.2にDRAMの基本回路原理図を示す．スイッチQ_sとコンデンサC_sよりなる1ビット**メモリセル**はマトリクス状に配置されている．各列には1個の**センスアンプ**とその両側に2個の**ダミーセル**および多数の1ビットメモリセルを持ち，それらはセンスアンプの両側に対称に配置されている．

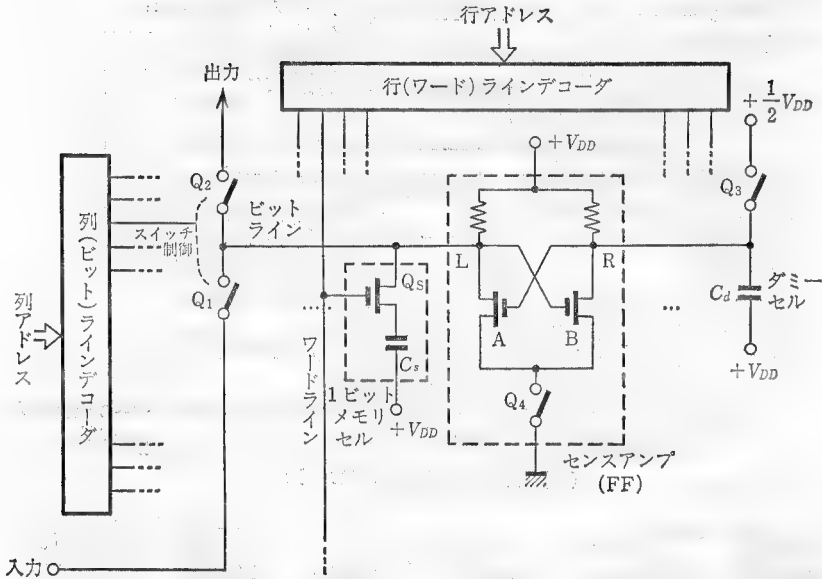



図 10.2 DRAM基本回路原理図(プリチャージ回路などは省略されている)

この回路でLレベルを書き込むときにはQ_1, Q_sをONにして入力端子にLレベルを加え，C_sの両端子間にV_{DD}をチャージする．読み出すときには，まず書き込みと反対側のダミーセルC_dに$\frac{1}{2}V_{DD}$をチャージする．つぎに，Q_sをONにしてC_sとC_dのチャージをそれぞれセンスアンプのL，R端子に加え，Q_4をONにして比較する．センスアンプ(FF)は少しでもゲート電圧の高いほうが先に動作するので，急速にAがON，BがOFFになる．したがってL端子はLレベルとなり，C_sには再びV_{DD}がチャージされる(リフレッシュ)．

実際のICでは，センスアンプはL端子とR端子間の50mV位の電位差をnsの速度で判定を行う性能を持っている．スイッチ類はMOSのトランスファゲートでつくられる．また，各ビットラインの非対称性を消すため，読み出しの前には左右の**ビットライン**の浮遊容量の電位を等しくしておく**プリチャージ**が行われる．実際のDRAMの内部構成とピン配置例を図10.3に示す．これらはいずれも電源は+5Vで，入出力はTTLコンパチブルである．これらのピン配

(a) 内部構成例 (1M×1ビット)

(b) ピン配置例 (1M×1ビット)　(c) ピン配置例 (256k×1ビット;日立HM50257)　(d) ピン配置例 (16k×4ビット；TI TMS4416，富士通MB81416など)

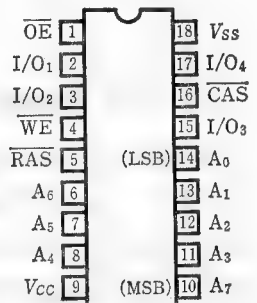
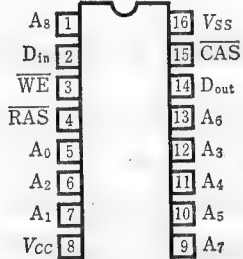
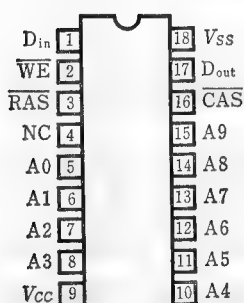
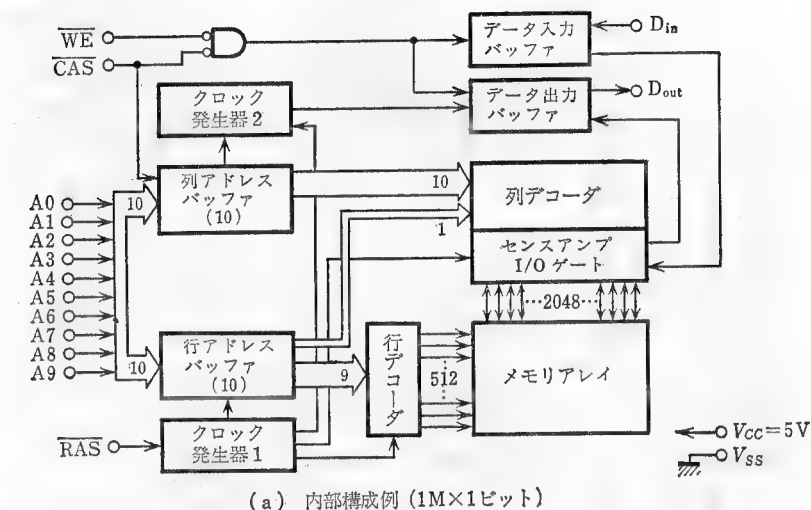



図 10.3 DRAM

置からわかるように同一容量のメモリでも同一アドレスに何ビットのデータが入れられるかというデータ構成法がいくつか異なるものがある．なお，大容量メモリ素子ではパッケージはα粒子によりデータが一時的に変ってしまう**ソフトエラー**を防ぐため，α線の出にくい材料でつくられる．また，反射によりα線を打ち消す工夫がなされることもある．

アドレスの指定は，**行アドレス**と**列アドレス**を分けて順次入れることにより

行われる．これにより，狭いアドレス幅(たとえば8ビットや10ビット)で広い(16ビットや20ビット)範囲のアドレスが指定できる．すなわち，少ないピン数，限られたパッケージサイズに大きな容量のチップを収容できる．このような方式を**マルチプレクスドアドレス方式**と呼ぶ．$\overline{\mathbf{WE}}$は write enable で，これがLレベルのときデータの書き込みができる．$\overline{\mathbf{RAS}}$は row address strobe で，これを用いて初めの8ビットを受けつけ，つぎに，$\overline{\mathbf{CAS}}$(column address strobe) を用いてつぎの8ビットのアドレッシングを行う．

10.2.2 動作タイミング

図10.4にはDRAMの動作タイミングを示す．まず，行アドレスをアドレス線にセットしておき$\overline{\mathrm{RAS}}$をLレベルに下げると，行アドレスが内部にラッチされる．それは，内部的に行デコーダでデコードされ，行ライン制御信号となる．つぎに同様に列アドレスをセットした後$\overline{\mathrm{CAS}}$をLレベルにすると内部的にアドレスバス上の列アドレスがラッチされ，列デコータでデコードされ，列ライン制御信号がつくられる．このとき，図10.4(a)のように$\overline{\mathrm{WE}}$をHレベルにしておくと読み出しとなり，列アドレス指定より一定時間後に出力データが$\mathrm{D_{out}}$端子に出てくる．これは$\overline{\mathrm{CAS}}$がLの間出力される．書き込み時には$\overline{\mathrm{WE}}$をLレベルにして列アドレスを指定すると，そのときの$\mathrm{D_{in}}$端子のデータがとり込まれ，記憶される．

$\overline{\mathrm{WE}}$と$\overline{\mathrm{CAS}}$は実際にはどちらか遅いほうの立ち下りで列アドレスとデータが読み込まれる．また連続書き込みの場合には，$\overline{\mathrm{WE}}$はLのままでよい．なお，端子にセットしたアドレスやデータは$\overline{\mathrm{RAS}}$や$\overline{\mathrm{CAS}}$を下げた後も指定された時間保持しておかなくてはならない．

単にメモリの**アクセスタイム**といえば$\overline{\mathrm{RAS}}$からのアクセスタイムのことで，$\overline{\mathrm{RAS}}$を落してから有効データが出てくるまでの時間である．これはDRAMではたとえば120nsである．しかしながら，$\overline{\mathrm{CAS}}$がある範囲より遅れてしまうとそれは無効となり，$\overline{\mathrm{CAS}}$からのアクセスタイムとして規定される．

また，**サイクルタイム**とは1つのデータをアクセスしてからつぎのアクセス

が行えるまでの時間である(図10.5参照)．DRAMの場合には，$\overline{\mathrm{RAS}}$からのアクセスタイムにRASプリチャージ時間を加えたものである．RASプリチャージ時間とは，リフレッシュのための時間で，DRAMの内部状態をもとに戻すのに必要な時間である．この間，$\overline{\mathrm{RAS}}$はHレベルにしておく．DRAMのサイ

(a) データの読み出し

(b) データの書き込み

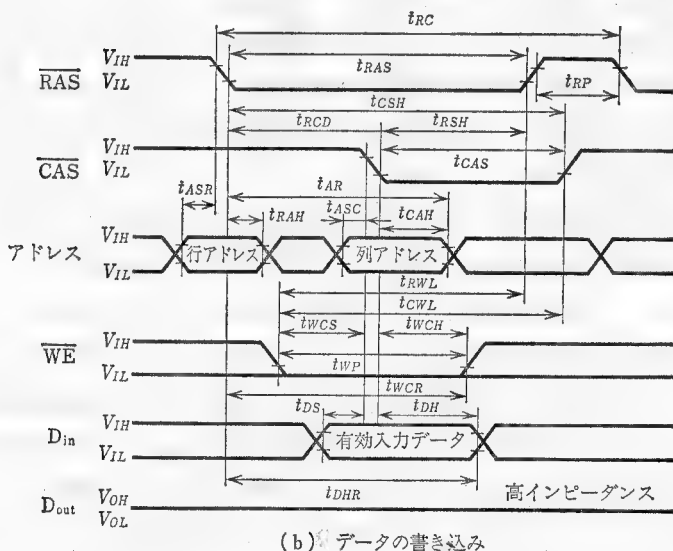
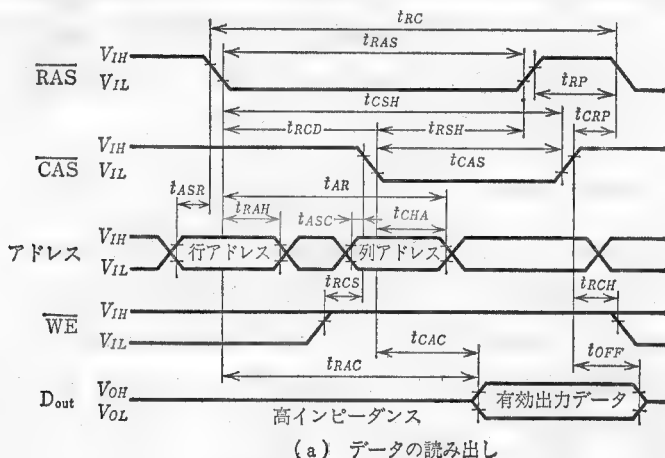



図 10.4　DRAMの動作タイミング

クルタイムはたとえば200 nsである．なお，DRAMの消費電力は256 kビットの例で動作時350 mW，待機時20 mWである．

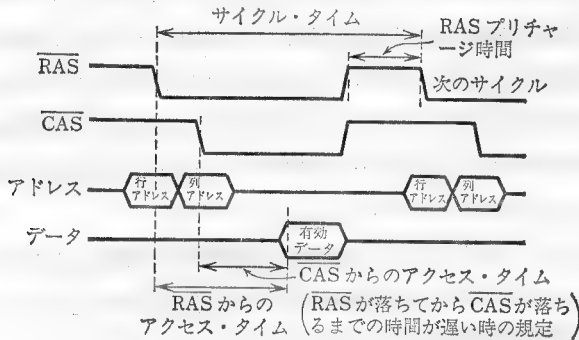



図 10.5 アクセスタイムとサイクルタイム

10.2.3 高速モード

一度行アドレスを内部のラッチにセットすると，$\overline{\text{RAS}}$をLレベルに固定したまま，$\overline{\text{CAS}}$により列アドレスを変えて読み込み，書き込みができる．このようにすると，行アドレスを毎回セットしなおす必要がなく，アクセスタイム，サイクルタイムを短くできる．また，$\overline{\text{RAS}}$の動きがないので，平均消費電力が小さくなる．このように，高速化のため，同一行アドレス内のアクセスに対しては一度行アドレスを与えた後は列アドレスの指定のみで，または自動的に連続した列アドレスへアクセスできる動作モードを持ったメモリ素子がつくられている．これらには，ページモード，ニブルモード，スタティックカラムモードなどがある．これらを図10.6により説明する．読み出しの場合を考える．

ページモードでは最初アドレスラインに行アドレスを乗せておいて$\overline{\text{RAS}}$をLレベルに下げて行を選択する．つぎに$\overline{\text{WE}}$を下げておいてから$\overline{\text{CAS}}$を下げて，列アドレス指定を行うと，最初のデータが読み出される．つぎに$\overline{\text{CAS}}$を一度上げてから再び下げ，列アドレス指定を行うとつぎのデータが読み出される．以下同様に列アドレスを与えておいて$\overline{\text{CAS}}$を上下させると連続的に同

一行内のデータアクセスが行える．このとき，1本の**ワードライン**が選択されたとすると，それにつながっているすべてのメモリセルが活性化され，それらのデータは各ビットライン上に現われる．したがって，列アドレスの指定のみによってデータが得られるのである．なお，RASプリチャージ時間は，同一ワードライン上のすべてのメモリセルの読み書きを行ったとしても最後に一回でよい．このような操作は，行アドレス指定により本の同一ページを開いたまま列アドレス指定により，そのページ内のいろいろな場所を指定する動作に似ていることからページモードという．

ニブルモード（4ビットモード）ではこれを簡略化して最初の列アドレス指定以降，計4ビットまでは列アドレスをセットせずとも$\overline{\mathrm{CAS}}$の上下のみで連続した4ビットの列アドレスのデータがアクセスできる．

スタティックカラムモードでは，$\overline{\mathrm{CAS}}$はLレベルのまま列アドレスのみを

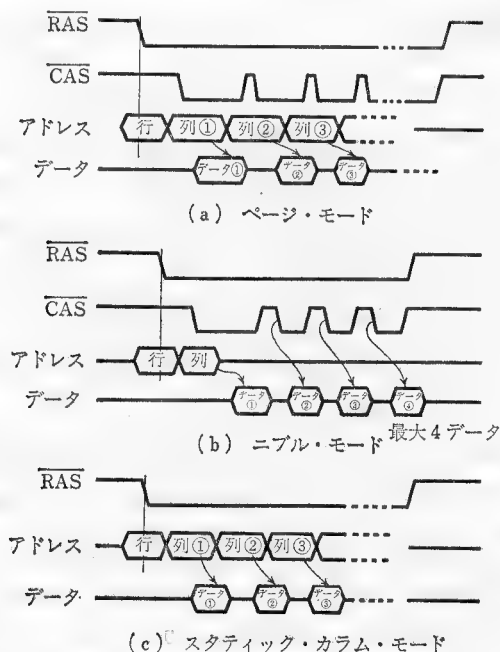



図 10.6　DRAMの高速動作モード（読み出し時）

変えることによりアクセスを行うことができるものである.

10.2.4 DRAMシステム

DRAM等のデータ構成は，たとえば256kビットといわれるものでは256k×1ビット，すなわちデータの入出力線1本でアドレスを256k種指定するタイプのほか，64k×4ビット，32k×8ビットなどのタイプがある．これらを用いて種々の形式のメモリシステムを構成できる．**DRAM**では記憶内容を保持しておくため，リフレッシュ動作が必要である．DRAMでは読み出し，もしくは書き込みを行うと，素子の内部で同一のワード線につながれたセルの再書き込み(リフレッシュ)が自動的に行われる．(意図的に)リフレッシュを行うためには図10.7に示すような動作タイミングで，2～8ms以内にすべての(または1個おきの)行アドレスを順に指定してやればよい．

図10.8には1MビットDRAMを8個並べて構成した1Mバイトメモリシステムを示す．左側のリフレッシュカウンタでは8msごとにすべての行アドレスを発生する．マルチプレクサMPX-1でリフレッシュカウンタと実際の行アドレスを切り換えている．さらに，MPX-2でこのアドレスと列アドレスを切り換えている．リフレッシュを行うときには，MPX-1はリフレッシュカウンタ側，MPX-2は行アドレス側に切り換えてリフレッシュカウンタを1ずつ000000000_2～111111111_2まで増加させていく．リフレッシュにはこの動作をこ

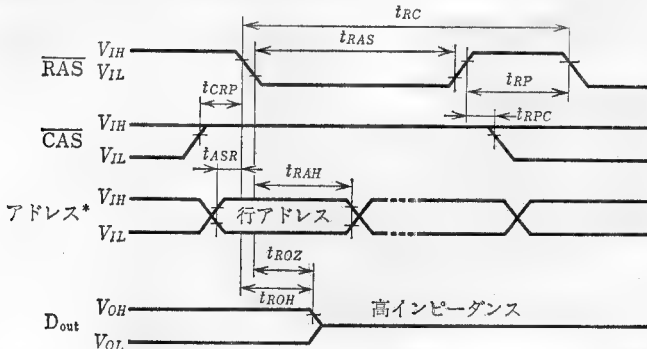



図 10.7 RASオンリリフレッシュサイクル（1MビットDRAMの例では行アドレスはA_0～A_8)

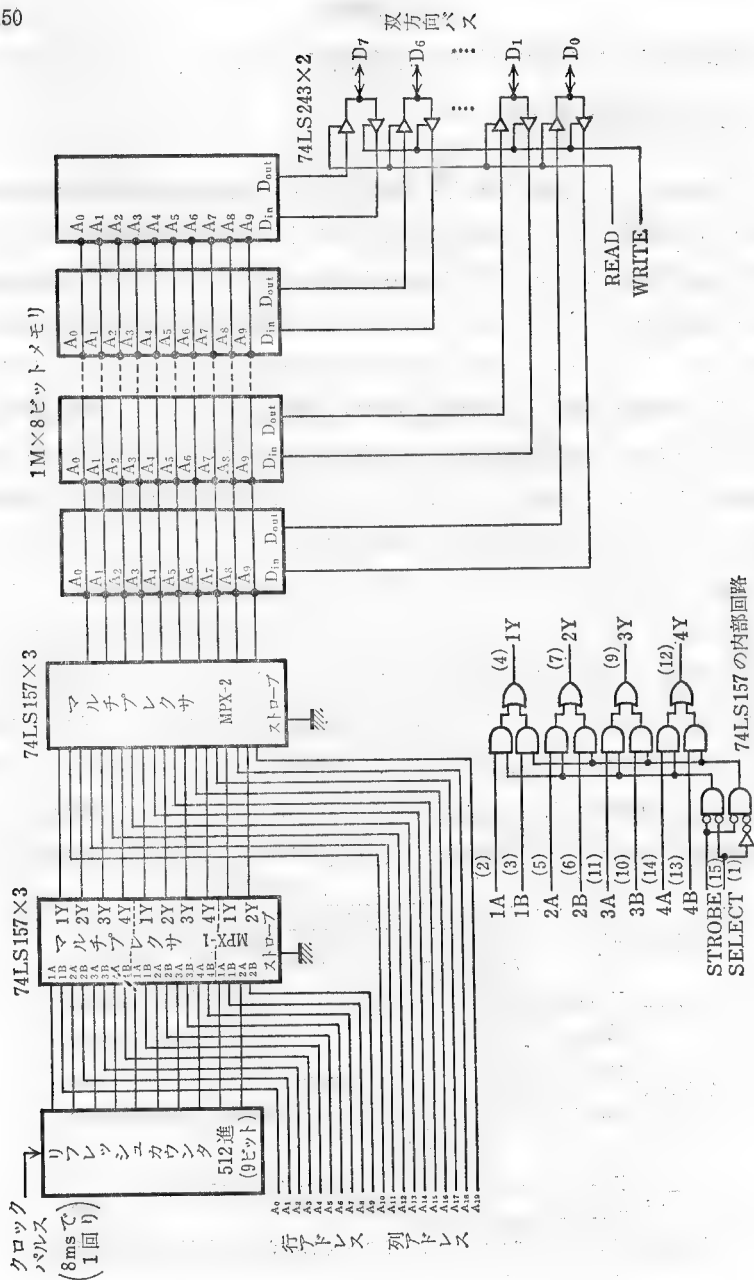



図 10.8 DRAMシステムの例(1M×8ビット)

ま切れに行う方法と，まとめて行う方法がある．

10.3 SRAM

SRAMには高速読み出し可能なnMOSと，低消費電力向きのCMOSがある．また，両者の長所を生かし，メモリセルをnMOSで，また周辺回路をCMOSで構成した素子もある．

CMOSのスタティックメモリセルは図10.9に示すフリップフロップが中心で，それに読み出し／書き込み用のトランジスタがつき，1セル6個のトランジスタで構成される．このセルにデータを書き込む場合，ワード線WをHレベルにすると，D上のデータがFFに書き込まれる．読み出しのときも同様にWをHレベルにしてD（または$\overline{D}$）の状態を調べればよい．読み出し，書き込みを行わないときにはWはLレベルにしておく．nMOSのセルも基本的には同じ構成であるが，小形化，低消費電力化のため，かつてのE/Dセルに代り，ポリシリコン高抵抗負荷（数十M～数百MΩ）を使用したE/Rセル（図10.10）が主流になってきた．図10.11に256kビットCMOS SRAMの内部構成を示す．また，図10.12には，そのピン配置，図10.13には動作タイミングを示す．

SRAMではプリチャージ時間が不要のため，アクセスタイムとサイクルタイ

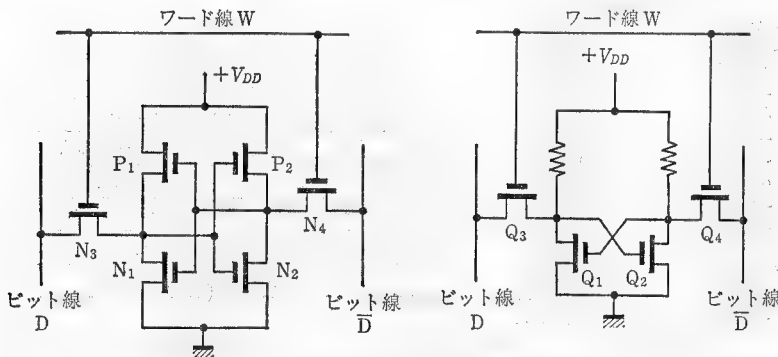



図 10.9 CMOSスタティックメモリセル回路　図 10.10 nMOSスタティックメモリセル回路

ムは等しくなる．これらはたとえば25～200 nsである．また，消費電力はCMOSでたとえば動作時30 mW/MHz，待機時1 mWで小さい．

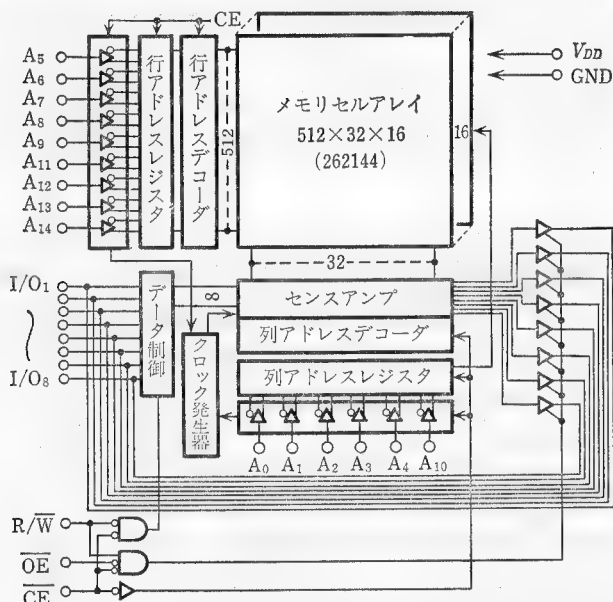
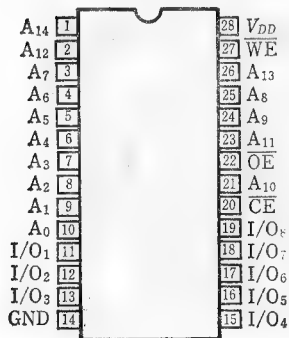



図 10.11　256 k(32 k×8)ビット CMOS SRAMの内部構成(東芝 TC 55257)

トップビュー(上から見た図)

ピン名称

A_0～A_{14}	アドレス入力
$\overline{WE}$	ライト・イネーブル
$\overline{OE}$	アウトプットイネーブル入力
$\overline{CE}$	チップイネーブル入力
I/O_1～I/O_8	データ入・出力
V_{DD}	電源端子（+5V）
GND	グラウンド

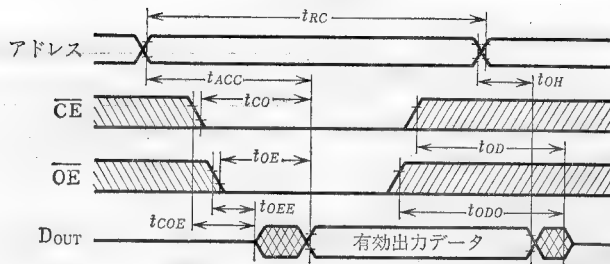
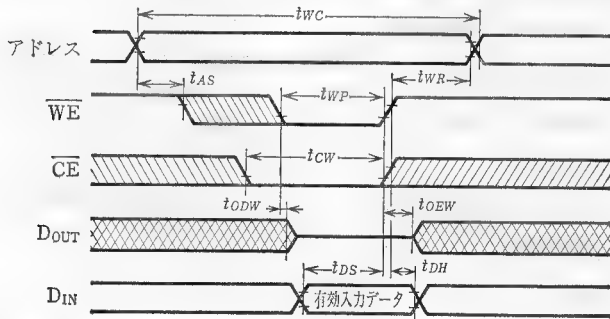
図 10.12　32 k×8 ビット SRAM ピン配置(東芝 TC 55257，日立 HM 62256)

（a） 読み出し（$\overline{WE}$=H, $\overline{CE}$=L, $\overline{OE}$=Lで出力）

（b） 書き込み（$\overline{WE}$=L, $\overline{CE}$=Lで入力, $\overline{OE}$はHが望ましい）

図 10.13 SRAMの動作タイミング

10.4 ROM

10.4.1 マスクROM

ウェハマスク工程時に情報を書き込んだ専用のマスクを使ってつくるROMでたとえば漢字フォントなど同一固定情報の入ったROMを大量につくる場合に適している．これはバイポーラでも不可能ではないが，通常MOS型が使用されるので，以下ではMOS型についてのべる．

マスクROMでは図10.14のように，マトリクス状に配列されたMOSの状態を変えて情報を記憶する．MOSの状態はつぎのような方法で変える．

（i） ゲートの酸化膜厚さを変えてしきい値電圧を変えるやり方．具体的に

は酸化膜厚さが薄いとしきい値電圧が低く，中間電圧でONとなり，それが薄いとしきい値電圧が高く，中間電圧でOFFとなる．なお，基板の不純物濃度を変えることによってもしきい値電圧を変えることができる．

(ii)　EMOSにするか，DMOSにするかにより，0V入力における電流の有無で判定できるようにする方法．

図 10.14　マスクROM

(iii)　MOSを接続するか，しないかで記憶する**コンタクト方式**もある．

状態の読み出しは，対応する駆動線をHレベルにしたときの読み出し線の電位により行う．マスクROMの読み出し速度(アクセスタイム)は，たとえば150～350nsであり，容量は4Mビットのものもつくられている．

10.4.2　PROM

PROM(programable ROM)は**FPROM**(field programable ROM)ともいわれ，ユーザが内容を自由に書き込めるものである．これには，情報を一度しか書き込めないものや，書き換え可能なものがある．

10.4.3　書き換え不可能形PROM

情報を一度しか書き込めない狭義のPROMである．図10.15に示す2つのタイプが代表的である．1つは同図(a)のように大電流でマトリクス交点のヒューズを溶断することにより情報を書き込んでいくもの，もう1つは同図(b)の

ように交点にある逆方向に接続されたダイオードの1つを，ブレークダウン電圧を加えて逆方向に大きな電流を流して破壊して短絡状態にして書き込んでいくものである．同図(c)のようにダイオードの代りにバイポーラトランジスタを用いるものも多い．最近の大容量高速PROMはこのようなバイポーラPROMが多く，アクセス時間は30〜50 ns程度と速い．

この原理を利用して**フィールドプログラマブル・ロジックアレイ**もつくられ

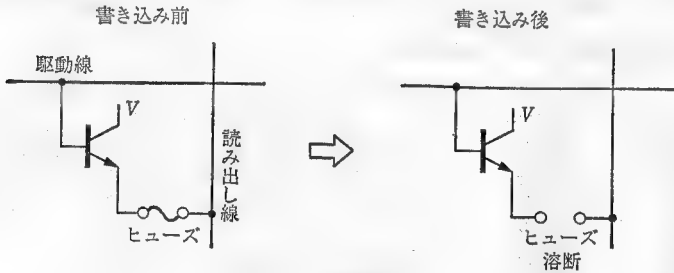
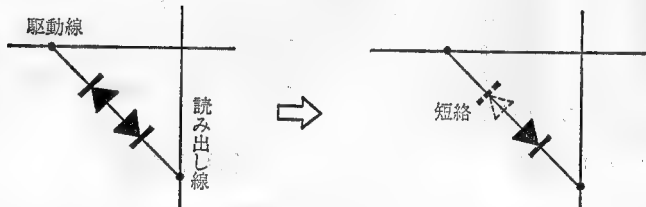
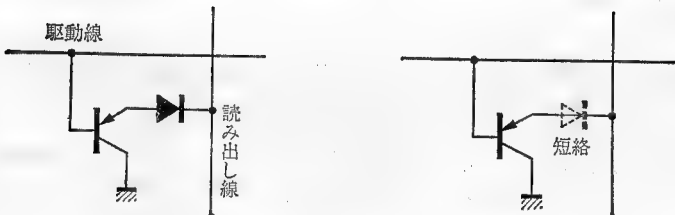



図 10.15 書き換え不可能形PROMの書き込み

ている．図10.16にその例を示す．同図(b)の機能図において，上半分はたとえば

$$P_0=I_0\cdot I_1\cdot \bar{I}_2\cdot I_5\cdot \bar{I}_{11}$$

のような積項をつくり，下半分では$\overline{\mathrm{CF}}$=Lのとき

$$F_0=(P_0+P_1+P_2)\quad：正論理の場合$$
$$F_0=\bar{P}_0\cdot \bar{P}_1\cdot \bar{P}_2\quad：負\quad〃$$

などのような論理式をプログラムできる．また，出力段のExclusive ORにより，出力の反転を得ることもできる．

(a) ピン配置　　(b) 機能図

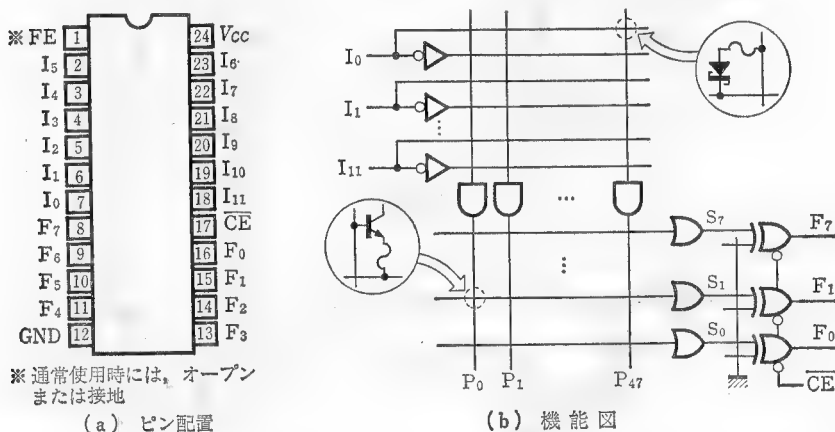



図 10.16　フィールドプログラマブル・ロジックアレイ(82S161，シグネティックス社)

10.4.4 EPROM

EPROM(erasable PROM)は一番よく使われる書き換え可能なROMである．しかしながら，RAMとは異なり，書き換え回数には制限があり，あまり多く書き換えることはできない．この代表的タイプとして，nタイプ**FAMOS**(floating gate avalanche injection MOS)についてのべる．

これは図10.17のようにMOSトランジスタの構造を基本としているが，酸化膜内部に浮いた形で多結晶シリコン(ポリシリコン)の**フローティング第1ゲ**

ートを持っている．その上に第2ゲートがある．第2ゲートに正の高い電圧を加えると，ドレインと基板のpn接合部で電子なだれ(アバランチェ)が起き，それが酸化膜の障壁を越えてホットエレクトロン(熱い電子)となってフローティング第1ゲートへ注入される．この電荷は放電するのに常温で100年以上かかるといわれており，この電荷があるとMOSのしきい値電圧を高くする．そのため，ソース，ドレイン間には電流が流れにくくなる．この性質を利用してROMができる．紫外線を照射すれば電子は励起されて基板へ戻るため消去できる．消去は10〜30分程度かけて15 W･sec/cm^{2}位の紫外線照射により行われるが，パッケージにはそのための石英の窓が上部についている．

EPROMは1Mビットのものまでつくられている．アクセスタイムは200/250 ns程度であり，書き込みは12.5Vのパルスで行われる．さらに，EPROMを利用したフィールドプログラマブル・ロジックアレイもある．

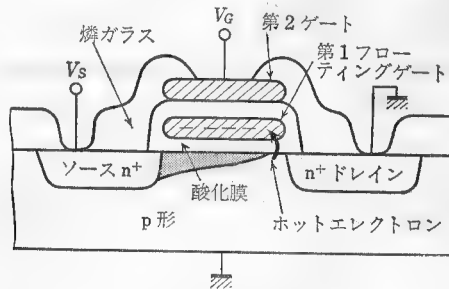



図 10.17 FAMOSの構造

10.4.5 EEPROM

EEPROM (electrically erasable PROM)は**EAROM** (electrically alterable ROM)とも呼ばれ，電気的に書き込め，かつ電気的に消去および再書き込み可能としたもので，シリコン酸化膜とシリコン窒化膜の2層構造を利用した**MNOS**(metal nitride oxide semiconductor)やフローティングゲート構造を利用したものなどがある．

図10.18には5V単一電源 2k×8ビットEEPROMの例を示す．また，その

動作モードを表10.1に示す．書き込みは$\overline{WE}$と$\overline{CE}$をL，$\overline{OE}$をHとすることにより開始される．アドレス入力は$\overline{WE}$か$\overline{CE}$の遅いほうの立ち下がりでラッチされる．I/Oピンのデータは$\overline{WE}$か$\overline{CE}$の立ち上りでラッチされる．データがラッチされると，このEEPROMは10 ms以内に選択されたバイトを自動的に消去し，そこに新しいデータを書き込む．

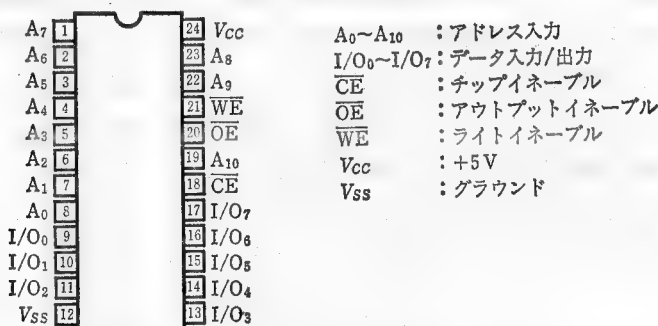



図 10.18　2k×8ビットEEPROM（5 V単一電源，MSM2816A，沖電気）

表 10.1　EEPROMの動作モード

$\overline{CE}$	$\overline{OE}$	$\overline{WE}$	モード	I/O	パワー
H	×	×	スタンバイ	High Z	スタンバイ
L	L	〃	読み出し	D_{OUT}	アクティブ
L	H	⊔	バイト書き込み	D_{IN}	アクティブ
L	H	H	読み出し／書き込み禁止	High Z	アクティブ

注：×；Don't care (H or L)

書き込み・消去のサイクルはバイト当り，1万回以上可能である．なおEEPROMは，64 kビットのものや，それを利用したICカードもつくられている．

従来のEEPROMは12～22 Vの高い制御電圧を必要としたが，現在主流のEEPROMは5 V単一電源で動作可能である．これは素子内部で高い電圧をつくっているからである．ただし現在でも全面消去には高い電圧を使うことがある．

10.5　マイクロプロセッサとの接続

多くのメモリ素子は**マイクロプロセッサ**(**MPU**)に容易に接続できる．図10.19にその簡単な例を示す．MPUとしては8ビットの代表的MPUであるZ80が使われている．

Z80を動かすクロック(ϕ)は2MHzを使用している．外部から意図的なMPUの制御($\overline{BUSRQ}$, $\overline{NMI}$, $\overline{INT}$, $\overline{WAIT}$)はリセットを除いて行っていない．電源投入時や必要な時にはMPUおよび並列インタフェースのリセット(RESET)を行う．

RAM，ROM，および並列インタフェースはA_0〜A_{12}およびCS(chip select; CEと呼ぶこともある)，$\overline{CS}$を用いてアクセスされる．それらの機能の選択は$\overline{WR}$(write; $\overline{WE}$と同じ)，$\overline{RD}$(read)，$\overline{OE}$(output enable)により行われる．ROMにはあらかじめROM書き込み器でプログラム等が書かれているとする．CSはそのチップ全体が選択され活性化されることを示す．ROMおよびRAMのアドレスバスへの接続から，ROM 2764は$A_{15}=0$かつA_{14}およびA_{13}には無関係，すなわち16進表現(H)で

①　0000〜1FFF$_H$

②　2000〜3FFF$_H$

③　4000〜5FFF$_H$

④　6000〜7FFF$_H$

の範囲でアクセスできることがわかる．実際にはこれら①〜④はROM内の物理的に同じ記憶場所を指しているので，0000〜1FFF$_H$と考えてよい．Z80をリセットすると0000から命令を読む．また，SRAM 6264は$A_{15}=1$, $A_{14}=0$, A_{13}=無関係であるから

①　8000〜9FFF$_H$

②　A000〜BFFF$_H$

のアドレスを持つ．さらに，並列インタフェース8255Aは$A_6=0$なる入出力

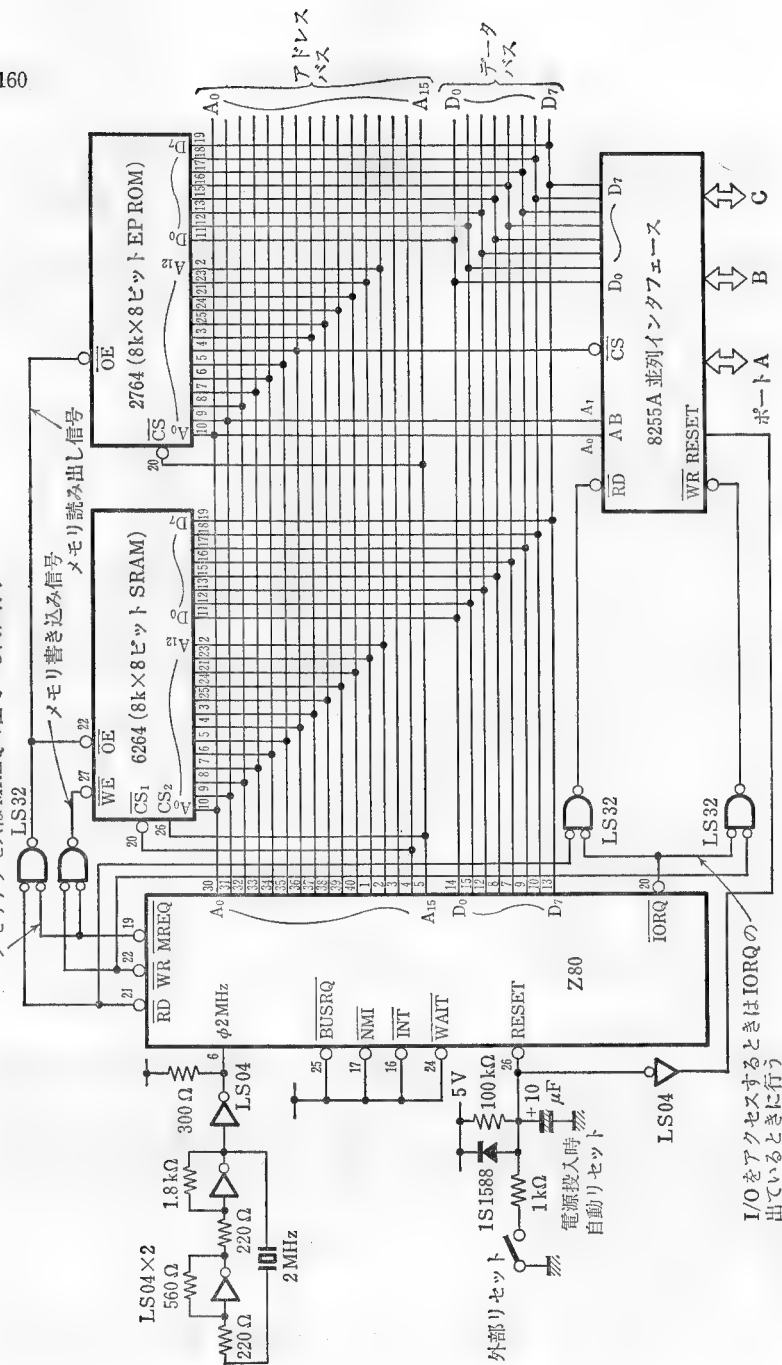



図 10.19 マイクロプロセッサとメモリの接続例

命令のときに選択され，その機能(ポートA, B, Cからのデータの入出力など)はA_0, A_1により決定される.

10.6　メモリ容量の拡張法

1個のメモリでは容量が不足する場合，いくつかのメモリ素子を使用して容量を増す．たとえば，256kビット(32k×8ビット)メモリを用いて128kバイトメモリを構成する場合を考える.

128kバイトのうちの1つを指定するには17ビットを要する．メモリチップのアドレス端子は15個(2^{15}=32k)であるから図10.20のように，$\overline{CS}$信号をゲート素子を用いてA_{15}，A_{16}よりつくればよいことがわかる.

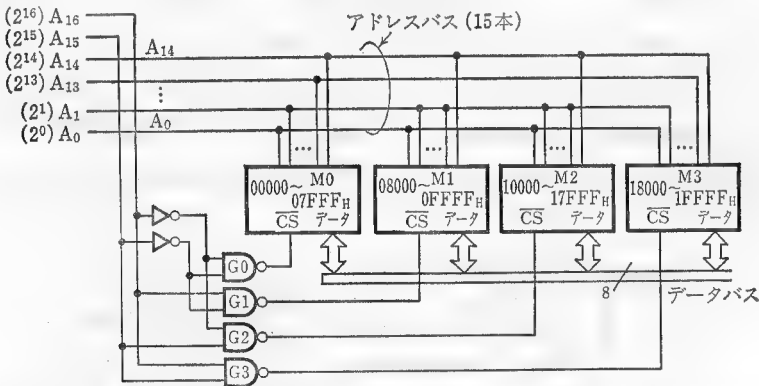



図 10.20　チップセレクトによるメモリ容量の拡張（32k×8ビットチップ4個で128kバイトメモリを構成する）

10.7　メモリシェア

メモリのアクセスタイムやサイクルタイムを素子の特性以上に速くしたいときに使われる方法に**メモリシェア**がある．これはメモリを数系統に分けて，それらを並列的にアクセスできるようにして見かけの速度を向上させる手法であ

る．分割は２系統にすることが多い．メモリシェアは，連続したアドレスは連続してアクセスされることが多いという事実を利用しているのである．

たとえば，アクセスタイム 80ns の 32k×8 ビット SRAM を用いて実効的に 40ns で連続した番地を順次アクセスできる 64k バイトメモリを構成することを考えよう．図 10.21 のように，偶数アドレスデータと奇数アドレスデータを別々のチップに入れるとする．また，アドレスカウンタを利用して連続した領域に偶奇振り分けて順にデータを書き込んでいくとする．メモリに与えるアドレスは 15 ビット（32k バイト）であるのでアドレスカウンタとして 15 ビットカウンタを使用する．

書き込みのときは図 10.22 のように偶数アドレスデータと奇数アドレスデータを交互に順次用意し，デマルチプレクシングした後，それらを偶数アドレスメモリと奇数アドレスメモリに同時に書き込む．この回路ではバッファ（ラッチ）が１個ずつしかないためバッファリングの時間が多少余分にかかっているが，バッファを２段にすればメモリサイクルの終りにつぎのデータのバッファリングを重ねて行うことが可能となり，速度が向上する．読み出しについても同様である．このような方法により，１度に２バイトのデータを読み出し／書

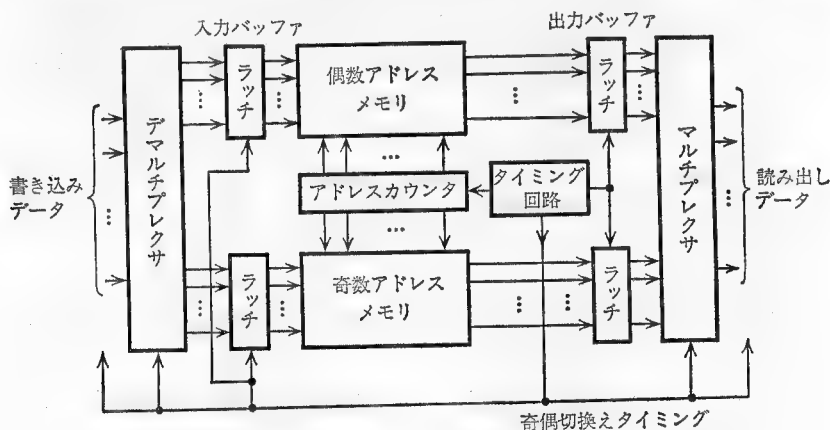



図 10.21　メモリシェアによるアクセス速度の向上

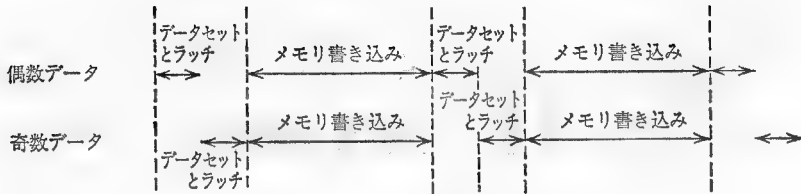



図 10.22　偶数奇数データの動作タイミング

き込みできるので実効的に速度を2倍にすることができる．さらに一般に n 系統のメモリ群を使えば最大 n 倍の速度となる．ただし，これは連続した領域を読み書きすることが多いときのみ有効である．

演習問題

[10.1]　DRAMとSRAMの違いについて述べよ．

[10.2]　つぎの語句を説明せよ．

(1)　マスクROM
(2)　PROM
(3)　EPROM
(4)　EEPROM
(5)　メモリシェア

[10.3]　1個のICメモリチップでは容量が不足するとき複数個のICメモリチップを使用する．そのとき，アドレスデータをどのように使ってアドレス指定を行えばよいか．

11　アナログ回路

11.1　A/D変換器

11.1.1　アナログ量のディジタル化

最近，ディジタル技術の進歩により，従来アナログ値で処理されていたものがディジタル的に処理されるようになってきた．そのとき，アナログ量とディジタル量との間の変換を行うものが**A/D変換器**，**D/A変換器**である．

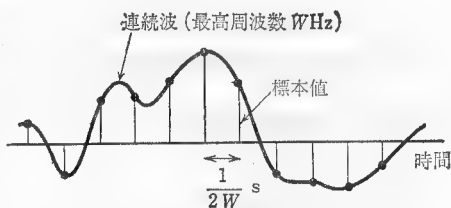



図 11.1　標本化

たとえば図11.1に示すようにWHzまでの信号を含むアナログ音声があったとする．これをディジタル量に変換することを考えよう．**標本化定理**により最高周波数の2倍，すなわち2WHzで標本化を行うと，もとの音声信号を損うことなく復元できる．このとき，各標本値はA/D変換器によりその大きさに応じたディジタル量に変換される．しかしながら，このときの**量子化レベル数**には限りがあり，そのため限られたビット数でアナログ量を表現することによる**量子化誤差**が生ずる（図11.2）．量子化誤差を小さくするためにはレベル数を増やし，分解能を大きくとる必要がある．

アナログ標本値をディジタル量に変換するものがA/D変換器であるが，A/D変換器として実用化されている形式には2重積分形，逐次比較形，並列比較形がある．その入力としては±5V，±10Vなど＋～－の範囲を許す**バイポーラ形**，および0～5V，0～10Vなど片側極性のみ許す**ユニポーラ形**がある．また，それらに応じて出力の形式にはバイナリ，オフセットバイナリ，2の補数，BCDなどがある．これらを表11.1～4に示す．

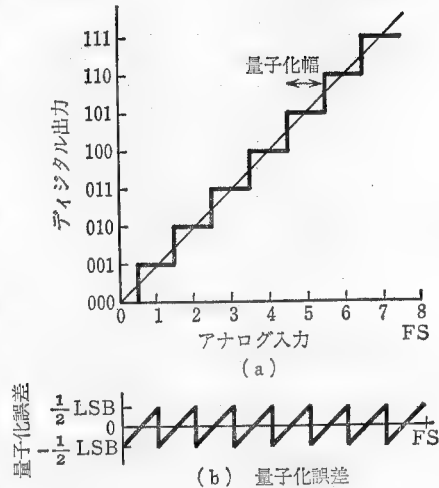



図 11.2 A/D変換器の特性

11.1.2 アパーチャタイムとサンプルホールド

量子化レベル数以外のA/D変換器

表 11.1 4ビット・バイナリコード表(FS＝フルスケール)

MSB	BIT 2	BIT 3	LSB	MSB BIT 2 BIT 3 LSB	
1	1	1	1	10.000×(8/16+4/16+2/16+1/16)=9.375V	FS-1LSB
1	1	1	0	10.000×(8/16+4/16+2/16+ 0)=8.750V	
1	1	0	1	10.000×(8/16+4/16+ 0 +1/16)=8.125V	
1	1	0	0	10.000×(8/16+4/16+ 0 + 0)=7.500V	3/4FS
1	0	1	1	10.000×(8/16+ 0 +2/16+1/16)=6.875V	
1	0	1	0	10.000×(8/16+ 0 +2/16+ 0)=6.250V	
1	0	0	1	10.000×(8/16+ 0 + 0 +1/16)=5.625V	
1	0	0	0	10.000×(8/16+ 0 + 0 + 0)=5.000V	1/2FS
0	1	1	1	10.000×(0 +4/16+2/16+1/16)=4.375V	
0	1	1	0	10.000×(0 +4/16+2/16+ 0)=3.750V	
0	1	0	1	10.000×(0 +4/16+ 0 +1/16)=3.125V	
0	1	0	0	10.000×(0 +4/16+ 0 + 0)=2.500V	1/4FS
0	0	1	1	10.000×(0 + 0 +2/16+1/16)=1.875V	
0	0	1	0	10.000×(0 + 0 +2/16+ 0)=1.250V	
0	0	0	1	10.000×(0 + 0 + 0 +1/16)=0.625V	1LSB
0	0	0	0	10.000×(0 + 0 + 0 + 0)=0.000V	

表 11.2　オフセット・バイナリコード表（4ビット）

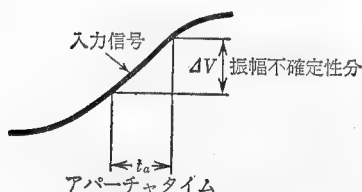
FS-1LSB	1 1 1 1	+4.375 V
FS-2LSB	1 1 1 0	+3.750 V
	1 1 0 1	+3.125 V
+FS/2	1 1 0 0	+2.500 V
	1 0 1 1	+1.875 V
	1 0 1 0	+1.250 V
+1LSB	1 0 0 1	+0.625 V
ゼロ	1 0 0 0	0.000 V
−1LSB	0 1 1 1	−0.625 V
	0 1 1 0	−1.250 V
	0 1 0 1	−1.875 V
−FS/2	0 1 0 0	−2.500 V
	0 0 1 1	−3.125 V
	0 0 1 0	−3.750 V
	0 0 0 1	−4.375 V
−FS	0 0 0 0	−5.000 V

表11.3　2の補数コード表（4ビット）

+FS-1LSB	0 1 1 1	+4.375 V
+FS-2LSB	0 1 1 0	+3.750 V
	0 1 0 1	+3.125 V
+FS/2	0 1 0 0	+2.500 V
	0 0 1 1	+1.875 V
	0 0 1 0	+1.250 V
+1LSB	0 0 0 1	+0.625 V
ゼロ	0 0 0 0	0.000 V
−1LSB	1 1 1 1	−0.625 V
	1 1 1 0	−1.250 V
	1 1 0 1	−1.875 V
−FS/2	1 1 0 0	−2.500 V
	1 0 1 1	−3.125 V
	1 0 1 0	−3.750 V
−FS+1LSB	1 0 0 1	−4.375 V
−FS	1 0 0 0	−5.000 V

図 11.3　アパーチャタイムと振幅不確定性分

表 11.4　3桁BCDコード表

10進数	MSD 10^2	DIGIT 2 10^1	LSD 10^0
9 9 9	1 0 0 1	1 0 0 1	1 0 0 1
⋮	⋮	⋮	⋮
8 6 4	1 0 1 0	0 1 1 0	0 1 0 0
⋮	⋮	⋮	⋮
7 7 7	0 1 1 1	0 1 1 1	0 1 1 1
⋮	⋮	⋮	⋮
5 0 0	0 1 0 0	0 0 0 0	0 0 0 0
⋮	⋮	⋮	⋮
2 2 2	0 0 1 0	0 0 1 0	0 0 1 0
⋮	⋮	⋮	⋮
1 0 0	0 0 0 1	0 0 0 0	0 0 0 0
9 9	0 0 0 0	1 0 0 1	1 0 0 1
⋮	⋮	⋮	⋮
1 0	0 0 0 0	0 0 0 1	0 0 0 0
⋮	⋮	⋮	⋮
1	0 0 0 0	0 0 0 0	0 0 0 1
0	0 0 0 0	0 0 0 0	0 0 0 0

の重要な特性は**アパーチャタイム**（**変換時間**）である．この時間内に1 LSB以上の入力電圧変動があると変換精度が不十分なものとなる．図11.3からわかるように，入力電圧 $V(t)$，振幅不確定性分 ΔV とアパーチャタイム t_a との間には

$$\Delta V=\frac{dV(t)}{dt}\times t_a$$

の関係がある．いま $V(t)$ として周波数 f Hz の正弦波 $V_0 \sin 2\pi ft$ を考えよう．dV/dt の最大値は零交差付近で起る．そのとき

$$\Delta V = \frac{d}{dt}(V_0 \sin 2\pi ft)|_{t=0} \times t_a = V_0 \; 2\pi f t_a$$

$$\frac{\Delta V}{V_0} = 2\pi f t_a$$

例として $f=1$ kHz の正弦波を12ビットの精度で A/D 変換する場合を考えよう．$2^{12}=4,096$ であるから

$$t_a = \frac{1}{2\pi f} \cdot \frac{\Delta V}{V_0} = \frac{1}{6.28\times10^3\times4\times10^3} \simeq 40\times10^{-9}\mathrm{s}$$

すなわち t_a は 40 ns というような高速性が要求される．このような短時間に精度のよい変換値を得ることは難しい．そのような場合，**サンプルホールド**回路が用いられる．これは，入力電圧値をコンデンサに一定時間保持する回路であり，A/D変換器はその間に変換を行えばよいことになる．図11.4にサンプルホールドIC回路の例を示す．

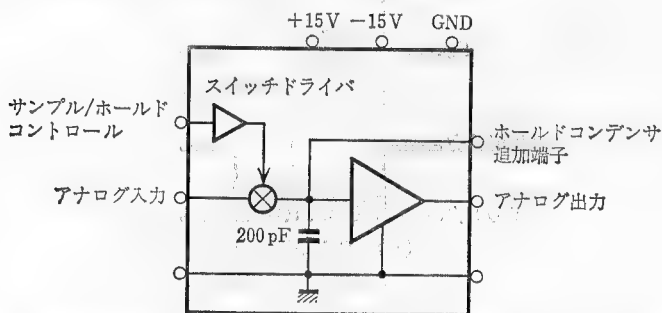



図 11.4　サンプルホールドIC(デイテル SHM-2，アパーチャタイム 10ns)

11.1.3　2重積分形A/D変換器

2重積分形A/D変換器は図11.5のように構成される．最初，入力電圧 e_i を一定時間 (t_0) 積分し，つぎに積分された電圧を減算する形でそれが初期値0に戻るまで基準電圧 e_r を積分する．この積分に要する時間が t であったとすると

$$e_i t_0 = e_r t$$

したがって

$$e_i=\left(\frac{e_r}{t_0}\right)t$$

より入力電圧e_iが求まる．t_0およびtはクロックパルスのカウンタで計測する．本方式は，同一の積分器で入力電圧の積分と基準電圧の積分を行うため，積分定数やクロックパルス周波数の変動に影響されにくいという特長をもっている．変換精度は積分時間t_0，tの精度で決まる．

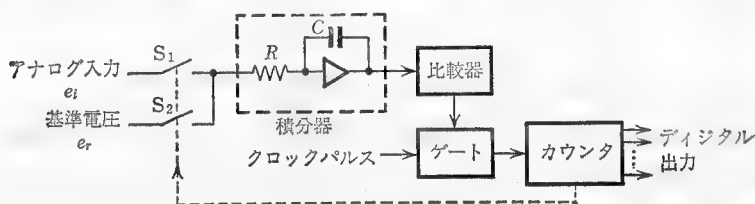
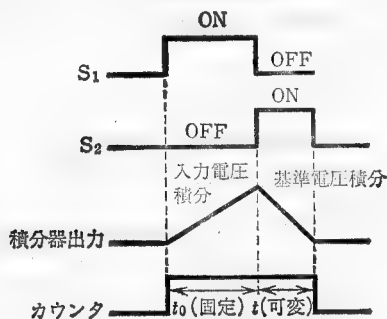



図 11.5　2重積分形A/D変換器

2重積分形のA/D変換器は分解能 8～16ビット，変換時間数ms～数百ms程度のCMOSでつくられたものが多く，低速ではあるが，精度を要するものに使われる．2重積分形は変換のアパーチャタイムが長い．そのため，入力のサンプルホールド回路を必要とする．また，標本化定理により，2重積分形は変換速度が遅いことから数百Hz以上の成分を含む信号のA/D変換は困難である．

11.1.4　逐次比較形A/D変換器

逐次比較形は内部に持つD/A変換器に仮の変換出力を順に入れていき，そ

の出力と外部からのアナログ入力との比較を行ってMSBより逐次，ディジタル出力を確定していく方法に基づく．図11.6にその構成図と変換の原理を示すタイムチャートを示す．まず，MSBに仮に1を立て，それをD/A変換する．その結果と，アナログ入力をコンパレータで比較し，アナログ入力のほうがD/A変換出力よりも大きければMSBに立てた1はそのままにしておく．もし小さければその1は0に変える．つぎにMSBはそのままで第2ビット目に仮に1を立て，同様にD/A変換を行い入力と比較して第2ビット目を確定する．このようにしてLSBまで確定していく．この方式に基づくA/D変換器は8～16ビットの分解能で変換時間は10～100 μs程度である．また，nMOS，CMOS，バイポーラなどでつくられる．この例を図11.7に示す．

(a) ブロック図

(b) 変換の原理

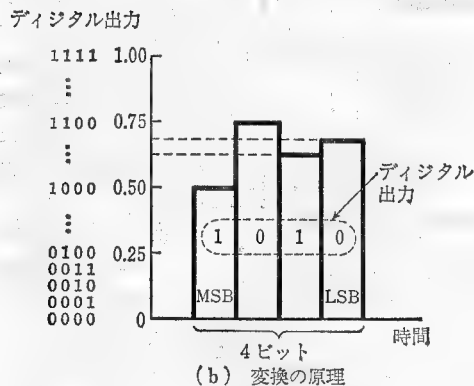
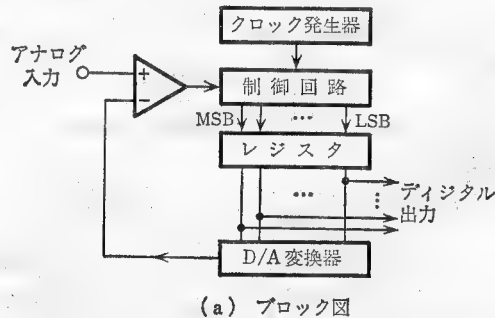
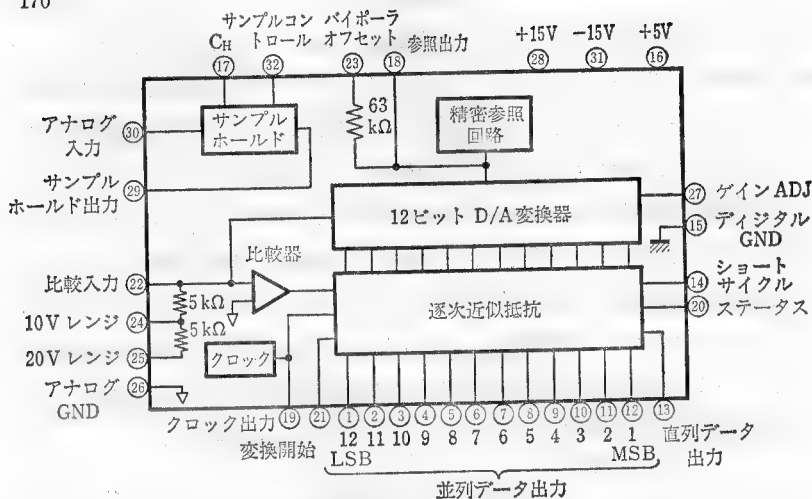



図 11.6　逐次比較形A/D変換器

図 11.7 12ビット逐次比較形A/D変換器の例(ディテル社ADC-HS12, 変換 9 μs, サンプルホールド内蔵)

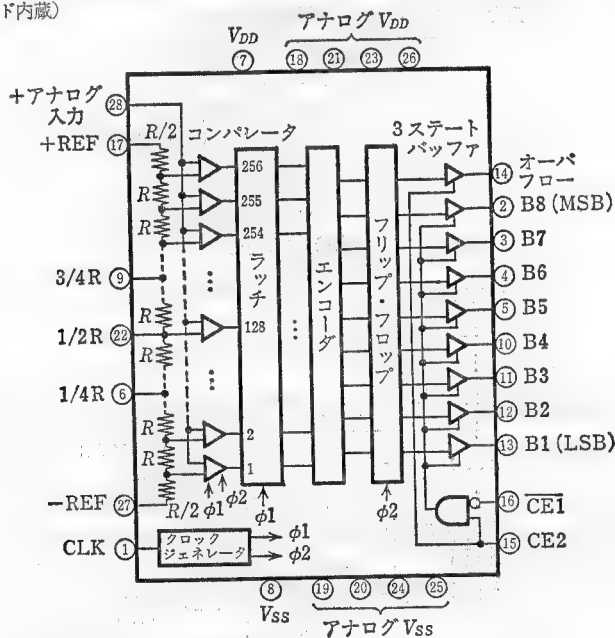



図 11.8 8ビット並列比較形A/D変換器(マイクロパワーシステムズMP7684, 15 MS/s, ただしMS/sとは 10^6 サンプル/sのこと)

11.1.5　並列比較形A/D変換器

並列比較形A/D変換器は，図11.8に示すように，変換レベル数だけの比較器を並べていろいろな基準電圧と入力電圧を一斉に比較する．基準電圧は抵抗回路により分圧されて，各比較器に入れられる．そして，比較の結果は2進数に変換される．この方式は**フラッシュタイプ**といわれ，一斉比較方式であるため変換速度が速く，サンプルホールド回路を必要としない．分解能は4～10ビット，変換時間は10～100 ns程度と非常に高速になっている．高速の信号を取り扱うことがしばしばあるため，CMOS，TTLのほか，ECLなどで構成される．

従来，数百～1000個のコンパレータを同一チップ内につくることは，なかなか難しかったが，マイクロエレクトロニクス技術の発展がそれを可能にしたのである．画像のような広帯域信号のA/D変換にはよく使われているが，他の分野でも主流になってくると思われる．なお，多くの比較器をつくるのが困難な場合は，一度に変換しないで，フルスケールを2～3段階に分けて変換する方式もある(**直並列変換**といわれる)．

11.2　D/A変換器

D/A変換器はディジタル値をアナログ量に変換するもので，A/D変換器よりも構成が楽である．図11.9にその基本となる**抵抗回路網**の例を示す．これ

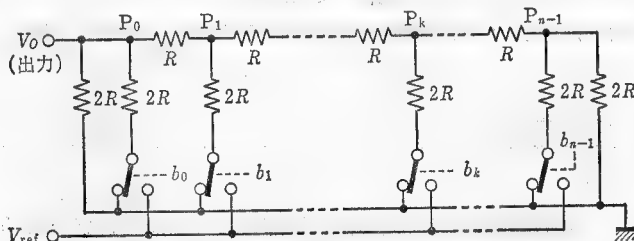



図 11.9　D/A変換器原理(抵抗回路網)

により，２進数が出力アナログ電圧値に変換される．図11.10に10ビット高速D/A変換器の例を示す．

D/A変換器の重要な特性はビット数とセトリングタイムである．**セトリングタイム**とは変換を開始してから出力が規定された誤差内に落ちつくまでに要する時間のことである．使用上留意しておかなければならない問題として，D/A変換器においては入力の変化に伴って出力が切り換わるとき，**グリッチ**といわれるスパイク状の雑音が出ることがある．これが問題となる場合には，出力をローパスフィルタに通したり，サンプルホールド回路でグリッチの出る時間には前の値が出力されるようにすればよい．

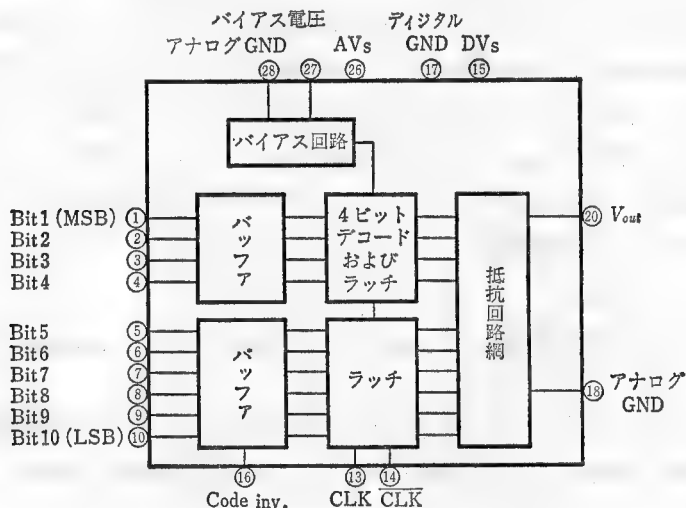



図 11.10　10ビットD/A変換器の例（デイテルDAC-330，160 MS/s，ECL入力）

11.3　オペアンプ

OP（オペ）アンプ（**演算増幅器**）単体は非常に増幅率の大きなアンプであるが，それに抵抗やコンデンサなどの受動素子によるフィードバックをかけて，精度のよい増幅器を構成する．たとえば，A/D変換器の入力範囲が±5Vのものを使用す

るとき，A/D変換すべきアナログ入力電圧のレンジ(範囲)が小さいと，十分なA/D変換精度が得られない．そのとき，図11.11のようにオペアンプを用いて増幅器を構成することにより，精度よく増幅・レンジ変換ができる．**反転増幅器**は入力電圧の符号が反転して出力される．また，**非反転増幅器**はそのままの符号で出力される．増幅率(ゲイン)の精度は抵抗値の精度で決まるため，精度を必要とする場合は金属皮膜抵抗を用いる．図11.12には代表的なOPアンプ741と318の開ループゲイン(負荷抵抗R_F等を何もつけないときのゲイン)と

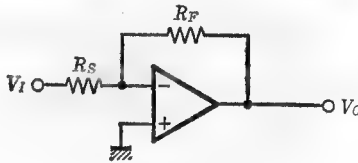
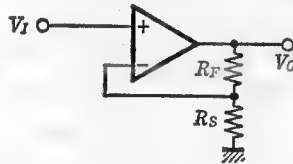
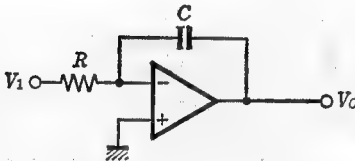
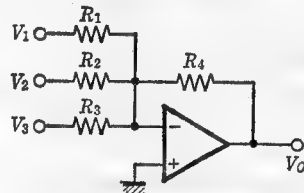
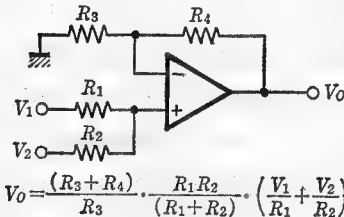
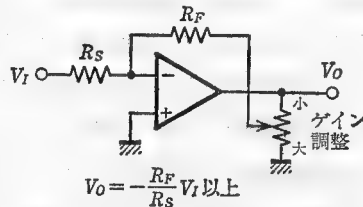



図11.11　オペアンプの基本的使用法

周波数の関係を示したものである．741はゲインが1となる周波数(帯域幅)は1MHz，スルーレート(下記参照)は0.5V/μsと低速である．318は同じく帯域幅15MHz，スルーレート50V/μsと高速である．

図 11.12　開ループ・ゲインの周波数特性(フィードバックをかけることにより点線のようになる)

これらに図11.11(a)のようにフィードバックをかけると図11.12の矢印で示すように，ゲインが押さえられ，点線のようなフラットな特性となる．フラットな範囲(正確にいえば-3dB，すなわち$1/\sqrt{2}$になるまでの周波数)は増幅器としての帯域幅を示すが，これは増幅率を上げると狭くなる．たとえば741は増幅率を100倍にすると帯域幅は10kHzとなってしまう．

スルーレートとは，急激に変化する入力が入ったとき，図11.13に示すように出力がどれ位の速度でそれに追従できるかを表すものであり，V/μsで表す．

図 11.13　スルーレート V_s [V/μs]

表11.5にはOPアンプの代表的品種と特性を示す．また，図11.14にはピン配置等を示す．356はオフセット調整端子を持ち，図11.14(d)のようにしてオフセット調整を行う．なお，**オフセット**とは入力が0Vのときの出力電圧のことで，0Vになるように合わせる．オフセット調整端子のないOPアンプも多い．また，高い周波数を扱うOPアンプでは**位相補償**用の端子を持っており，そこに外付けのコンデンサ等をつけて高域の特性の改善をはかる場合もある．なお，一般にOPアンプは非反転よりも反転で使用したほうが特性はよい．

図 11.14 代表的OPアンプのピン配置等

表 11.5 OPアンプの特性

型　名	動作電圧	帯　域（代表値）	スルーレート（代表値）	特　徴
4558	±4〜±18V	3.0 MHz	0.5V/μs	2回路入り．汎用バイポーラOPアンプ．かつての標準品741の広帯域化
324	3〜30V 単一電源	1 MHz	0.5V/μs	4回路入り．単一電源のため非反転で使用，LM 324，μA 324 などの型名を持つ．
356	±5〜±18V	5 MHz	12V/μs	BI-FET(入力段がともに接合型FET)
TL 084	±4〜±18V	3 MHz	13V/μs	4回路入り．汎用BI-FET
318	±5〜±18V	5 MHz	70V/μs (min, 50V/μs)	高速広帯域OPアンプ
LH0032C	±5〜±18V	70MHz	500V/μs	高速広帯域OPアンプ，ハイブリッド形
μPC252A				低入力リーク電流(代表値0.1pA，25°C) MOS-FET入力．pAオーダの微小電流の計測可
LM308				入力バイアス電流10nA以下．直流微小信号の計測などに使用
M4250	±1V以上			低消費電力(10μAでも動作可)，NS社
CLC220	±15V	200MHz	8,000V/μs	高スルーレート，コムリニア社

11.4　コンパレータ

アナログコンパレータは，２つのアナログ信号の比較を行い，どちらが大きいかを１，０の２値に変えて出力するものである．したがって，一方に一定の基準電圧を入力し，他方にアナログ信号を入力すると，図11.15に示すごとくアナログ信号の２値化ができる．

図 11.15　コンパレータ出力

コンパレータももともとOPアンプの一種であるが，OPアンプの出力がプラス・マイナスの両方向に振れるのに対し，コンパレータ出力は片方向(H, L)である．また比較に際しての応答特性が改善されている．コンパレータは外付け抵抗なしの開ループ状態で用いる．

表 11.6　コンパレータの特性

型　名	標準電源電圧	応答時間
LM 339	+5V(±15Vも可)	1.3μs
LM 319	±15V(+5Vも可)	80ns
LM 361	±10V(±15Vも可)	
LM 360	±5V(±6Vも可)	14ns

表11.6にはコンパレータの特性例を示す．また，図11.16にはコンパレータの使用法を示す．これらを５Vまたは±５Vで使用する場合，TTLを直接駆動することができる．

11.5　マルチプレクサおよびデマルチプレクサ

図11.17に示すように，多数のアナログ入力をスイッチを切り換えて１本の線に絞って出力するものがアナログ**マルチプレクサ**である．逆に，１本の線に乗っている情報をいくつかのラインに切り換えて分配するものを**デマルチプレ**

(a) 使用例　　(b) ヒステリシス特性を持たせたコンパレータ

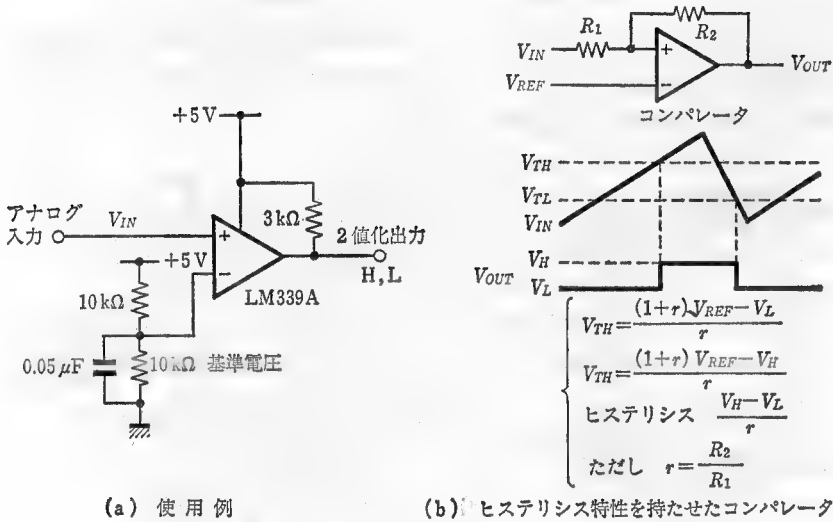



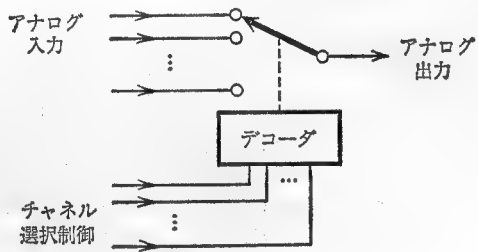
図 11.16 コンパレータの使用法

図 11.17 アナログマルチプレクサ

クサと呼ぶ．これらはCMOSで構成されることが多い．

アナログマルチプレクサは，たとえば多数のアナログ値を1個のA/D変換器でディジタル化したいというような場合に用いられる．

アナログマルチプレクサは，外部からディジタル制御信号を入力してスイッチを順次切り換えて，いくつかのアナログ信号の1つを選んで出力する．したがって，スイッチが閉じられてから対応するアナログ入力が出てくるまでに多少の時間遅れを生ずる．この時間遅れを**セトリングタイム**と呼ぶ．高速度で使用時にはこのことを注意しなければならない．

図11.18には，4チャネル・**アナログスイッチ**をマルチプレクサとして使用した例を示す．また，図11.19には8チャネル・アナログマルチプレクサを示

す．この場合，チャネル指定のためのデコーダが内蔵されている．

アナログスイッチの一応用例として，図11.20に差分積分器を示す．積分極性切換え信号により，このOPアンプは反転積分器または非反転積分器として働き，それらの積分値の差が出力される．

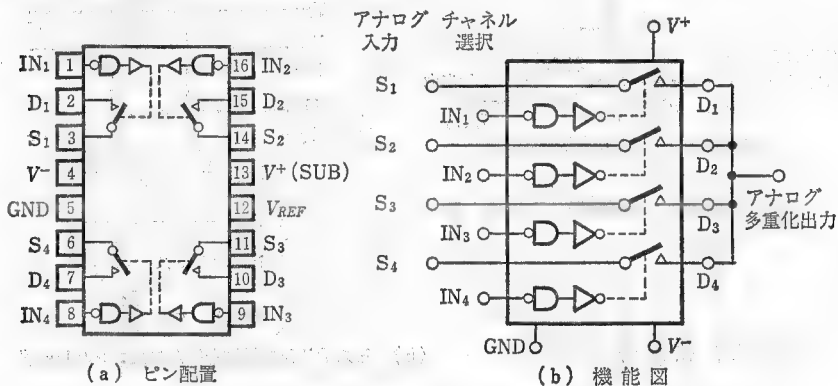



図 11.18　4チャネル・アナログスイッチIC(DG 201A，MAX 331，マキシム社等，電源電圧 ±4.5V〜±18V)

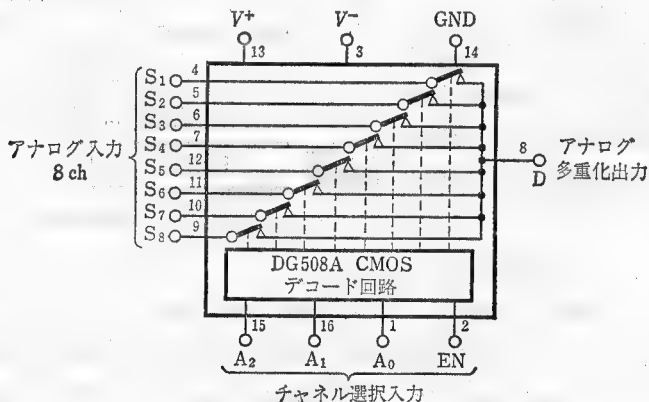



図 11.19　8チャネル・アナログマルチプレクサIC(DG 508A，マキシム社等)

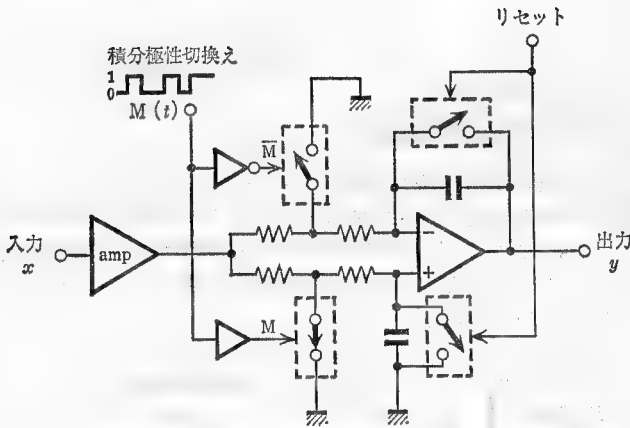



図 11.20　差分積分器(極性切換え信号により，入力信号$x(t)$が反転積分もしくは非反転積分され，それらの積分の差が出力$y(t)$となる).

演習問題

[11.1]　A/D変換器として実用化されている方式3つをあげ，その各々について簡単に説明せよ.

[11.2]　20 kHz までの周波数を含む音声を標本化定理にしたがって標本化し，かつ各標本値を12ビットでA/D変換すると，毎秒何ビットのデータとなるか.

[11.3]　つぎの語句を説明せよ.

(1)　OP アンプ

(2)　アナログコンパレータ

(3)　スルーレート

付　　　録

A.1　ゲートによる発振回路

クロックパルスなどのために正確な発振器をつくりたいときには，図A.1に示すような**水晶発振回路**が用いられる．同図(a)ではバッファなしのHS-CMOSを用いている．この回路にバッファつきのHS-CMOS(74HC04など)を使用すると，増幅度が大きすぎて発振が不安定となる．同図(b)はTTLによる水晶発振回路である．いずれも10^{-5}オーダの周波数安定度が達成できる．

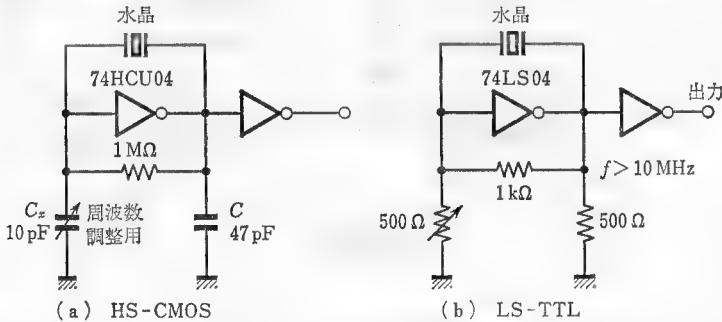



図 A.1　インバータによる水晶発振回路

図A.2にはHS-CMOS ICゲートのみによる発振回路を示す．水晶発振回路ほど安定ではないが，広い範囲の発振周波数が得られる(図A.3)．R_pはG_Aの入力保護抵抗でCのチャージによって生じる過電圧・ラッチアップを防止する．R_pは10 kΩ～500 kΩとする．発振周波数はおよそ

$$f=\frac{1}{2.2CR}\quad \text{[Hz]}$$

で与えられる．

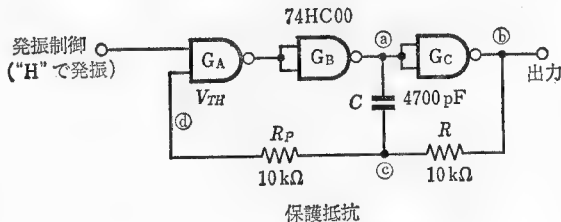



図 A.2　HS-CMOSゲートによる発振回路

図A.4にはTTLゲートによる発振回路を示すが，この場合は発振周波数はあまり変えられない．

図A.2の回路では発振制御入力が“L”レベルのときは，ⓐⓑⓒⓓは各々“L”，“H”，“H”，“H”で安定しており，発振しない．Cにはⓒ側が＋の電荷が蓄積されている．発振制御入力が“H”レベルになったとすると，ⓐⓑⓒⓓは各々“H”，“L”，“H”，“H”となる．しかしながら時間がたつと，Cの電荷はRを通して放電するから，ⓓの電位は下っていき，G_Aのいき値電圧V_{TH}より下ったところで各ゲートの状態は反転し，ⓐⓑⓒⓓは各々“L”，“H”，“L”，“L”となる．その後，今度はRを通してCに電荷が

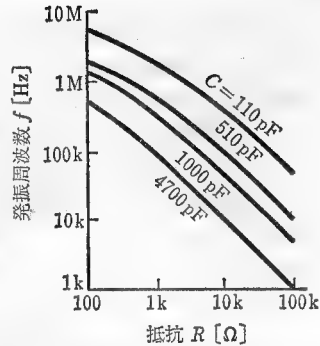



図 A.3　図A.2の発振周波数例

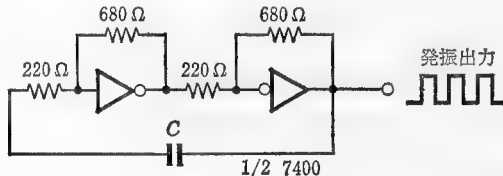



図 A.4　TTLゲートによる発振回路

蓄積されるからⓓおよびⓐの電圧は上昇していき，V_{TH}を越えところで各ゲートの反転が起こる．このようにして発振が起こる．

A.2　発光ダイオード

発光ダイオード(**LED**；light emitting diode)は順方向に電流を流すと赤や緑の光を出し，表示素子や光伝送用光源としてよく使われる．図A.5にLED駆動の基本回路を示

（a）直列スイッチング　（b）並列スイッチング

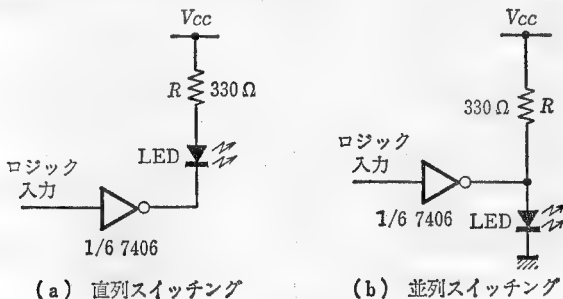



図 A.5　LEDの基本駆動方法(7406はオープンコレクタ)

す．

これを利用して，数字表示器である**7セグメント表示器**をBCD入力により駆動することを考えよう．7セグメント表示器は図A.6のような7個のLEDバーセグメントを発光させることにより数字を表示する．その真理値表を表A.1に示す．

この論理式は

表 A.1　バーセグメント表示回路の真理値表

2進入力 A	B	C	D	デコーダ出力 Z_a	Z_b	Z_c	Z_d	Z_e	Z_f	Z_g	表示
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	0	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	0	0	1	1	9

$$
\left.
\begin{aligned}
\overline{Z}_a &= \overline{A}\overline{B}\overline{C}D+\overline{A}B\overline{C}\overline{D}+\overline{A}BC\overline{D}\\
\overline{Z}_b &= \overline{A}B\overline{C}D+\overline{A}BC\overline{D}\\
\overline{Z}_c &= \overline{A}\overline{B}C\overline{D}\\
\overline{Z}_d &= \overline{A}\overline{B}\overline{C}D+\overline{A}B\overline{C}\overline{D}+\overline{A}BCD+A\overline{B}\overline{C}D\\
\overline{Z}_e &= \overline{A}\overline{B}\overline{C}D+\overline{A}\overline{B}CD+\overline{A}B\overline{C}\overline{D}+\overline{A}B\overline{C}D\\
&\quad +\overline{A}BCD+A\overline{B}\overline{C}D\\
\overline{Z}_f &= \overline{A}\overline{B}\overline{C}D+\overline{A}\overline{B}C\overline{D}+\overline{A}\overline{B}CD+\overline{A}BCD\\
\overline{Z}_g &= \overline{A}\overline{B}\overline{C}\overline{B}+\overline{A}\overline{B}\overline{C}D+\overline{A}BCD
\end{aligned}
\right\}
\qquad (A.1)
$$

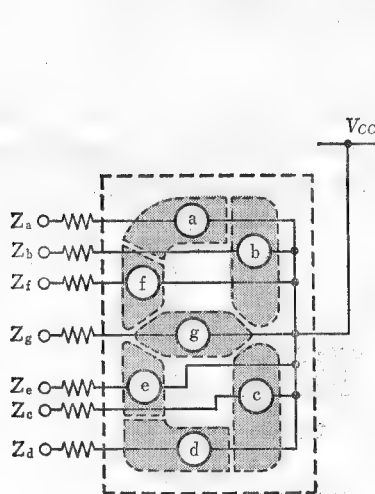



図 A.6　7セグメント表示器

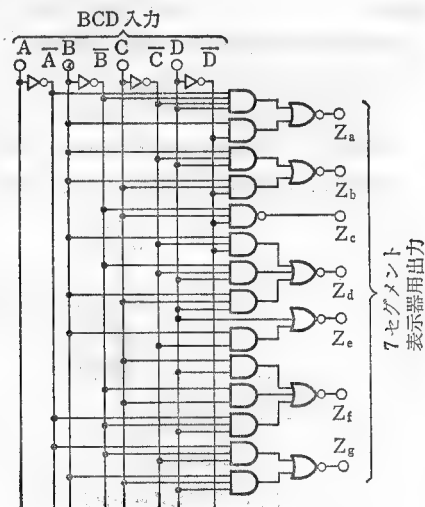



図 A.7　7セグメント表示デコーダ

で与えられる．冗長な入力1010～1111を用い，カルノー図により簡単化を行うと

$$\left.\begin{aligned}
&\overline{Z}_a=\overline{A}\overline{B}\overline{C}D+B\overline{D}\\
&\overline{Z}_b=B\overline{C}D+BC\overline{D}\\
&\overline{Z}_c=\overline{B}C\overline{D}\\
&\overline{Z}_d=B\overline{C}\overline{D}+\overline{B}\overline{C}D+BCD\\
&\overline{Z}_e=B\overline{C}+D\\
&\overline{Z}_f=\overline{A}\overline{B}D+\overline{B}C+CD\\
&\overline{Z}_g\overline{A}\overline{B}\overline{C}+BCD
\end{aligned}\right\}\qquad(A.2)$$

出力

Vcc　f　g　a　b　c　d　e

16　15　14　13　12　11　10　9

f g a b c d e

BI/

B C LT RBO RBI D A

1　2　3　4　5　6　7　8

B　C　ランプテスト　RB出力　RB入力　D　A　GND

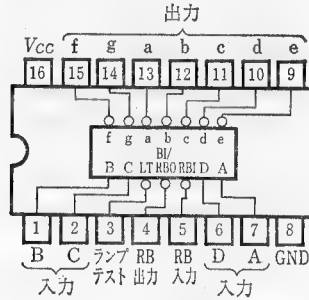
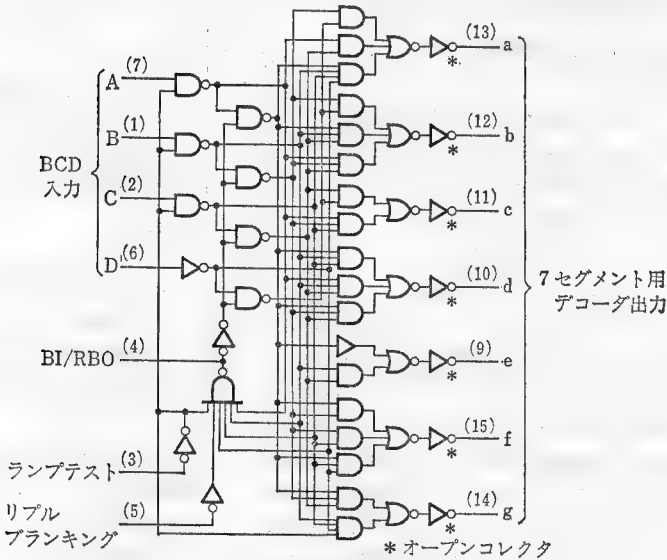
入力　　入力

（a）ピン配置

（b）内部等価回路

図 A.8　BCD入力7セグメントデコーダ／ドライバ(7446, 7447, 74 LS 47)

となる．これを論理回路で表すと図A.7のようになる．図A.8にはBCD入力7セグメントデコーダ／ドライバMSI 7446, 7447, 74LS47を示す．また，その使用例を図A.9に示す．同様の素子に7448, 7449, 74246～74249, 74347がある．

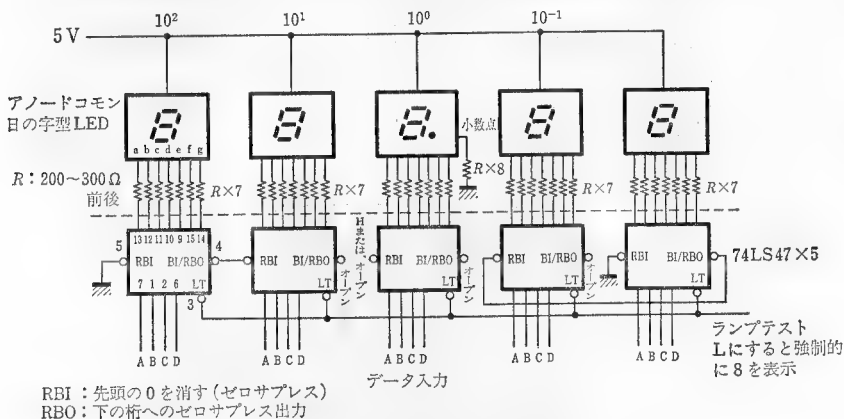



図 A.9　BCD入力7セグメント表示回路

A.3　RS-232C シリアルインタフェース

図 A.10　モデムインタフェース規定個所

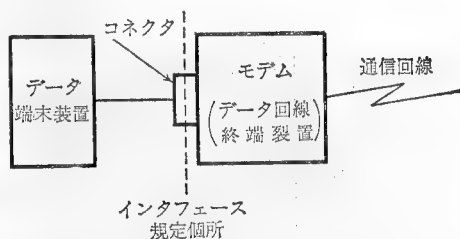
電話（アナログ）回線を用いてデータを送る場合，ディジタルデータとアナログ信号の交換を行う必要がある．そのときに用いられる変換器が**MODEM**(modulator and demodulator)である．CCITT (International Telegraph and Telephone Consultative Comittie；国連の機関）勧告V 24, JIS C 6361およびEIA（アメリカ電子機械工業会）規格の**RS-232-C**では，コンピュータなどのデータ機器とMODEMとの接続規格（図A.10参照）を定めている．現在ではMODEMとのインタフェースに限らず，広く一般にコンピュータと端末の接続のための**シリアルインタフェース**としてRS-232Cは用いられている．

RS-232Cの電気的な規格は表A.2のように定められている．これを満たす**レシーバ，ドライバ**ICを図A.11に示す．これらは図A.12のように使われる．レシーバSN75189Aは，電源電圧は5V単一でよいが，ドライバSN75188およびドライバとレシーバが対で入っているSN751701は±12Vの電源を必要とする．そのため，図A.11(d)のようにDC-DCコンバータを内蔵し，+5V単一電源で動作するドライバ/レシーバもある．

表 A.2　RS-232Cインタフェースにおける電気的規格

項　　目	特　　性
データ伝送速度	200 k ビット/s以下
負　　荷	直流抵抗…3～7 kΩ (3～15 V に対して) 容量…2500 pF 以下
信号源電圧 (送信の場合)	±5～15 V (3～7 kΩ の負荷に対して) 開放電圧…25 V 以下
受信信号の識別 (受信の場合)	"0"…+3 V 以上 (ON) "1"…−3 V 以下 (OFF)

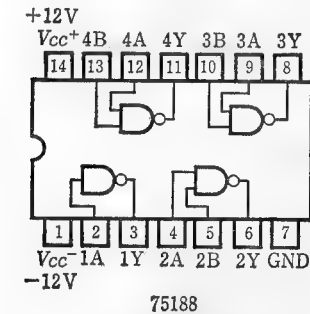
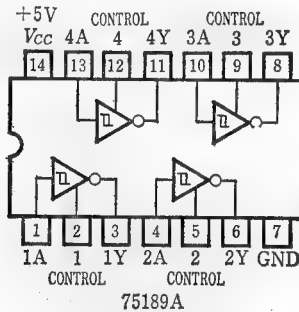
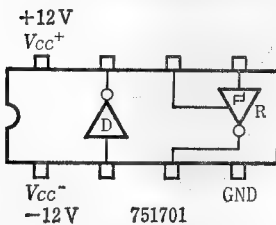
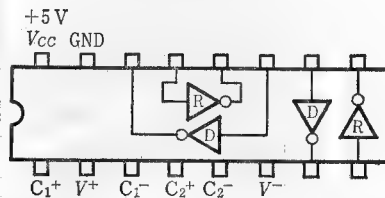
RS-232Cにおける信号線には図A.13のようなものがある．このうち，実際に直列データが乗るのはRD($\overline{R_XD}$；受信データ)およびSD($\overline{T_XD}$；送信データ)各1本のみで，他は制御用信号である．

(a) ドライバ (出力は ±9 V以上)　(b) レシーバ

(c) ドライバ/レシーバ　(d) DC-DCコンバータ内蔵型ドライバ/レシーバ

図 A.11　RS-232C用ドライバ/レシーバIC

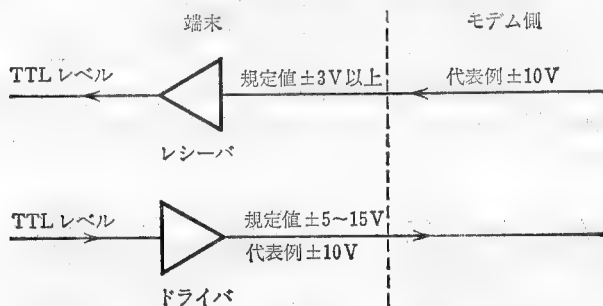
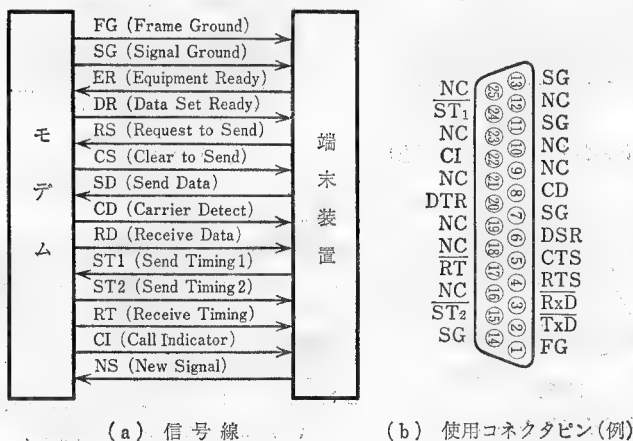
このような信号を発生するための**USART**(universal synchronous asynchronous receiver transmitter；汎用同期非同期送受信器)ICの代表的なものに8251Aがある．これを図A.14に示す．

A.4　フォトカプラ

2つのデータ機器間のグランド電位(GND)は必ずしも一致しない．そのような機器間でインタフェースを行う場合，GND同士を直接つなぐとその間に電流が流れ，それが雑音となったり，素子を破壊することがある．そこで，直流的に機器間を絶縁し，データのみを送れるようにするインタフェース素子として**フォトカプラ**がある．

図A.15に種々のスピードの代表的なフォトカプラ素子を示す．いずれも，送信デー

図A.12　インタフェースの電圧

(a) 信号線　　(b) 使用コネクタピン(例)

図A.13　RS-232Cインタフェース信号線

タで素子内蔵のLEDが発光し，それを低速のものはフォトトランジスタで，また高速のものはフォトダイオードで受光するようになっている．これらの使用例を図A.16に示す．使用上の注意として，高速のものは素子に接して電源ラインに0.01 μF～0.1 μFのバイパスコンデンサ(パスコン)を挿入しないと動作が不安定になることである．バイパスコンデンサとしては，周波数特性がよく，実効抵抗の低いセラミックコンデンサなどがよい．

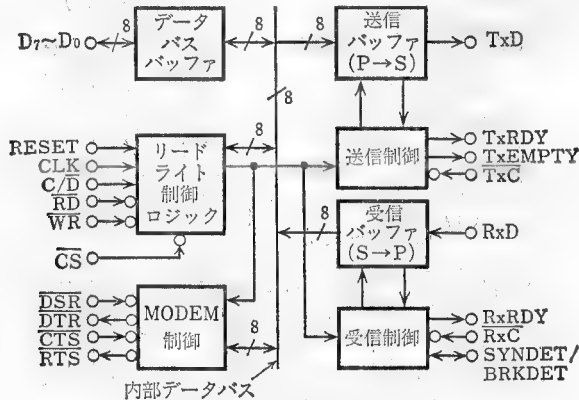
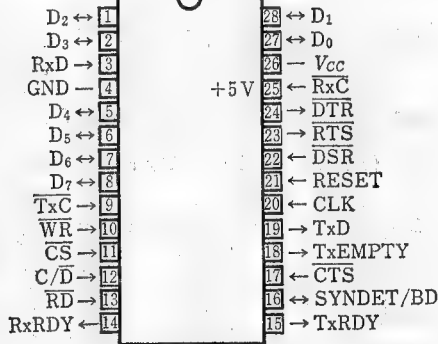



図 A.14　直列インタフェース(USART)用IC(8251 A)

真理値表（c）

入力	イネーブル	出力
ON	H	L
OFF	H	H
ON	L	H
OFF	L	H

真理値表（d）

入力	出力
H	L
L	H

真理値表（e）

入力	出力
H	L
L	H

真理値表（f）

入力	イネーブル	出力
ON	L	L
OFF	L	H
ON	H	Z
OFF	H	Z

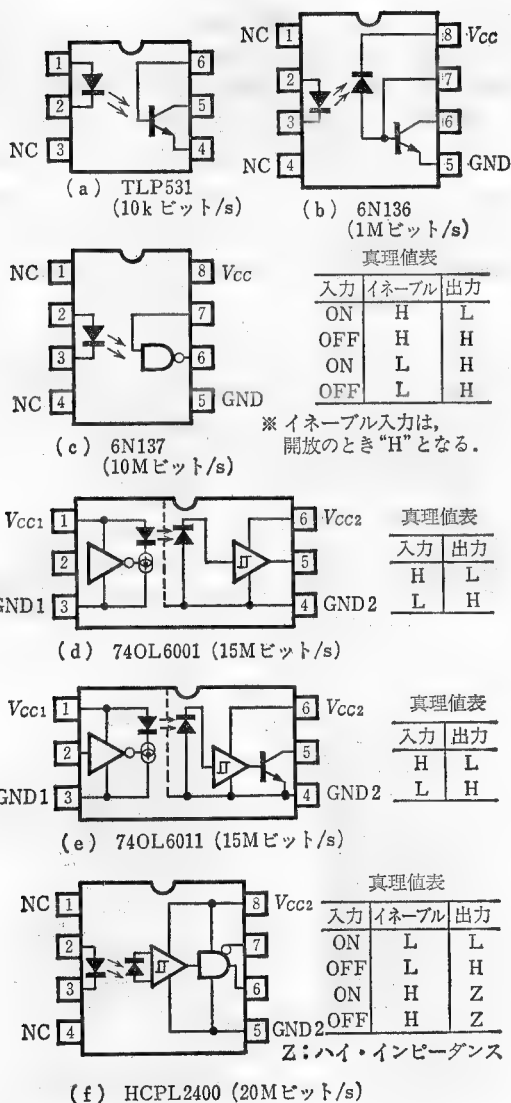



図 A.15 フォトカプラ素子（カッコ内は応答速度，電源電圧 (V_{CC}) はいずれも +5 V）

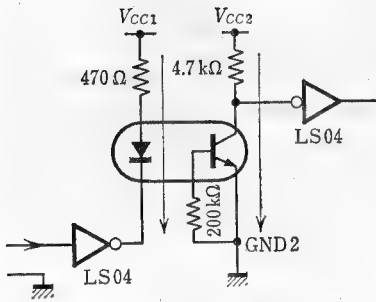
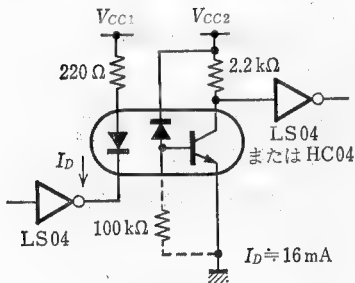
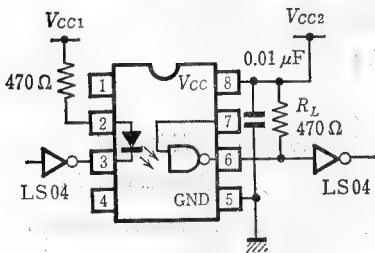
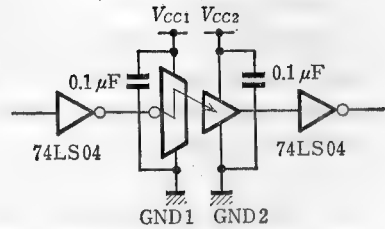
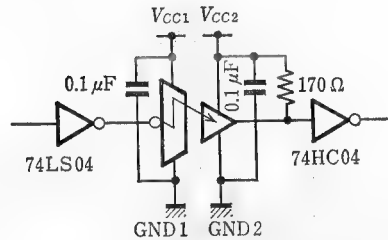
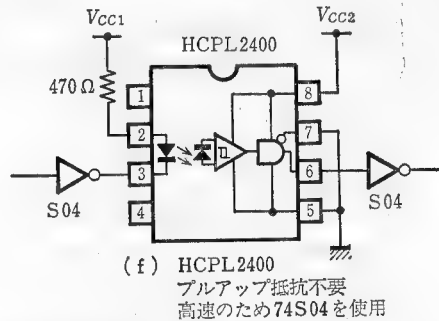



図 A.16　フォトダイオード使用例

A.5　トランジスタの伝播遅延時間と少数キャリア

トランジスタの時間応答を求めるためキャリアの振舞いをもう少し詳しくのべる．エミッタからベースへ流れ込んだ電子(ベース領域では少数キャリア)はベース領域を通過してコレクタに達するのであるが，その間は主として拡散によって移動する．したがっ

て，その動作は時間遅れを持つ．それを1次遅れ系としてモデル化すると，図A.17(a)のような回路ではベース電流のステップ状入力($0 \to i_0$)に対して同図(b)のsのような応答をする．この立ち上り時間を短縮するためにはベース電流を同図(c)のように，最初は大きく，つぎにコレクタ電流が飽和に達したら小さくi_0にすればよい．これを順方向**過励振**(forward overdrive)という．これにより，トランジスタのON時の伝播遅延時間が短縮される．ここでベース電流を過励振後，もとに戻したのは，トランジスタの飽和を深くしたままにしておくとつぎにのべるようにOFF時の蓄積時間が大きくなり，そのためOFF時の伝播遅延時間が大きくなってしまうからである．

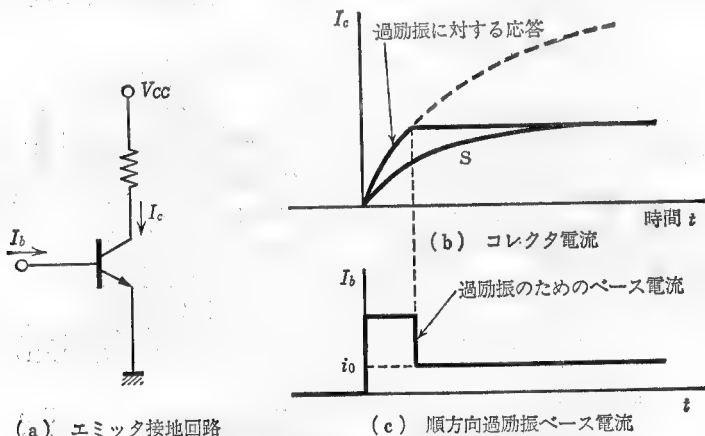



図 A.17　順方向過励振によるON時伝播遅延時間の短縮

つぎに，トランジスタの遮断時の振舞いについて考えてみる．いまトランジスタは飽和になる直前，すなわちコレクタ・ベース接合が順方向になる直前の状態にあるとする．ここでベース電流が急に0になったとする．このとき，エミッタからベースに注入された少数キャリアが拡散によりコレクタへ達するまでコレクタ電流は流れ続ける．したがって，トランジスタ遮断時もコレクタ電流は一定の時定数を持って減衰する．この立ち下り時間を短縮するためにはベース電流を0とするだ

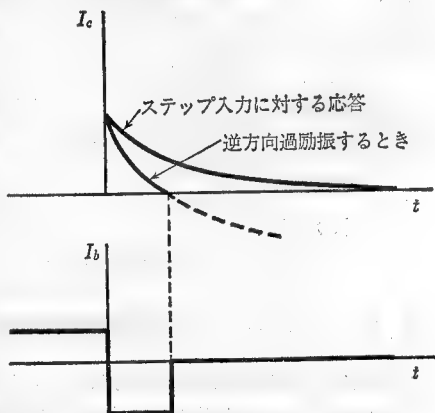



図 A.18　逆方向過励振によるOFF時伝播遅延時間の短縮

けでなく負に過励振して，すなわち一時的に負の電流をベースに加えて少数キャリアを引き出してやればよい．これを逆方向過励振(reverse overdrive)という(図A. 18参照)．

つぎに飽和状態にあるトランジスタの遮断時の振舞いを考えよう．飽和状態ではコレクターベース間の接合も順方向バイアスされるので，エミッタのみならずコレクタからもベース領域へ少数キャリアが注入され，その一部はエミッタへ達する．これはいわば**逆方向トランジスタ**といえるものが形成されたと考えることができる．したがって飽和状態にあるトランジスタは各々非飽和状態にある通常の(順方向)トランジスタと逆方向トランジスタを重ね合わせたものとみなすことができる．この飽和状態にあるトランジスタを遮断するためには，まず逆方向トランジスタを遮断して飽和状態を脱し，さらに順

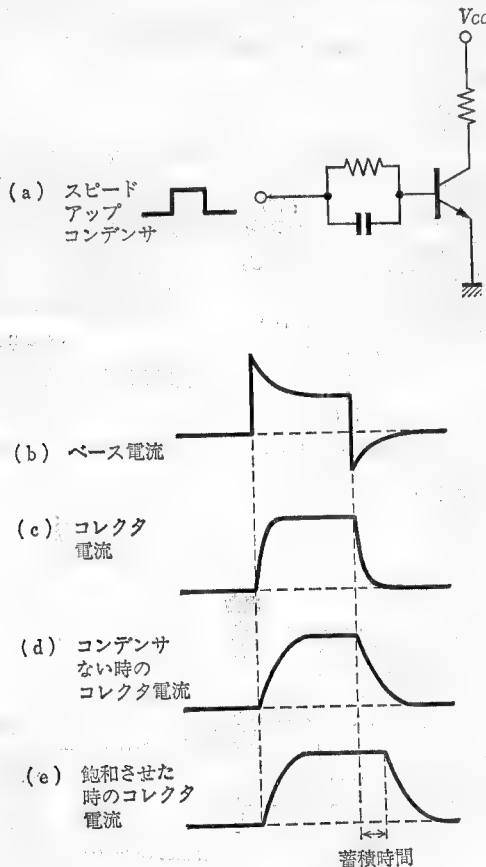



図 A. 19　スイッチング速度

方向トランジスタを遮断しなければならない．すなわち，飽和状態にあるトランジスタの遮断時間は逆方向トランジスタの遮断時間と順方向トランジスタの遮断時間の和となる．この前者の時間は**蓄積時間**(storage time)といわれる．

過励振は図A.19(a)に示すような**スピードアップ・コンデンサ**で簡単に行うことができる．同図はまた，その効果(b～d)，および飽和させた時の蓄積時間など，スイッチング時間に関連する事項を図示している．

A.6 IIL

バイポーラはMOSトランジスタに比較して高速であるが，製造工程が複雑で消費電力も素子の占有面積も大きいという欠点を持っていた．これを解決する回路として，1972年西独のIBMとオランダのフィリップス社から**IIL**(integrated injection logic；I^2Lとも書く)が発表された．

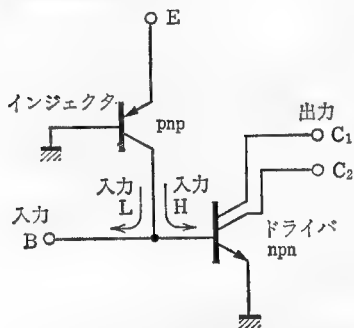



図 A.20 IIL基本回路(インバータ)

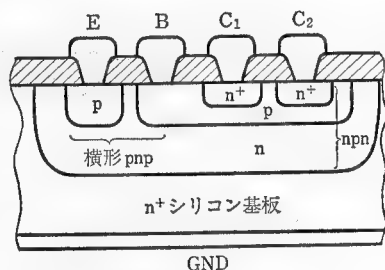



図 A.21 IILの基本構造

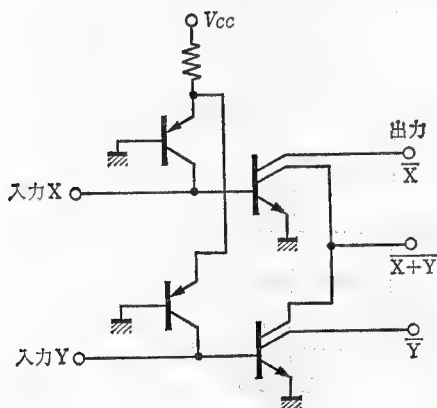



図 A.22 IILのNORゲート

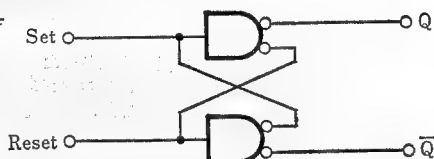



図 A.23 IILによるフリップフロップ

IILは図A. 20および図A. 21に示すように，pnpとnpnトランジスタを相補的に用いて回路を構成したものである．**インジェクタ**と呼ばれるpnpトランジスタは入力がLのとき前段のnpnトランジスタ(トライバ)の負荷抵抗として働くとともに，入力がHのときドライバであるnpnトランジスタのベース駆動電流源として働く．基本的にTTLが多入力1出力回路であるのに対し，IILは1入力多出力回路である．構造的には図A. 21に示すようにインジェクタのpnpトランジスタは横形で構成され，そのコレクタ領域とnpnトランジスタのベース領域は共用され，さらにpnpトランジスタのベース領域とnpnトランジスタのエミッタ領域が共用された複合構造をなしている．これにより小さくつくることができるとともに，バイポーラ・トランジスタに必要な素子間分離領域が不要な構造となっており，一層の高集積化が可能となっている．IILのNORゲートの例を図A. 22に示す．また，フリップフロップの例を図A. 23に示す．IILの特徴は高集積密度，低消費電力であり，たとえばTTLに比較して10倍程度の高集積化が実現されている．また消費電力は1〜10 μW/ゲートとCMOS素子より少ない傾向にある．また電源電圧も3Vぐらいで動作可能であり，電池電源を用いる応用面に適している．製造工程も標準的なバイポーラプロセスと同じでよく，バイポーラアナログ集積回路と同一基板上に共存も可能である．しかしながら，逆形pnpトランジスタ構造を使用したことにより標準的な伝播遅延時間は10〜100 ns/ゲートと遅く，とくにCMOS素子と比べ必ずしも優れているとはいえず，劣勢にある．

演習問題解答

1 章

[1.1] 各々入力 V_i が0Vのときトランジスタは遮断状態となる．したがって，そのとき V_0 は(a)，(b)ともに5Vとなる．一方，入力が各々10V，2Vのときにはベース電圧はともにおよそ0.7V(pn接合の順方向電圧)となるので，ベース電流 I_B は各々 $\dfrac{(10-0.7)\text{V}}{5\text{k}\Omega}=1.8\text{mA}$，$\dfrac{(2-0.7)\text{V}}{3\text{k}\Omega}=0.43\text{mA}$ となる．いま，$\beta=100$ であるので，不飽和状態ならば I_C は各々180mA，43mAとなるが，(a)では負荷抵抗による電流制限のため $I_C=70\text{mA}$(飽和状態)となる．そのときの出力電圧は各々0.2V，2.8Vとなる．

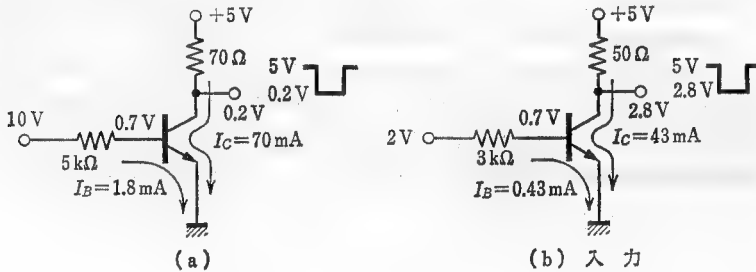



図 P.1

[1.2] 明らかに

$$\beta=\beta_1\cdot\beta_2$$

このように，ダーリントン接続では増幅率 β を大きくできる．

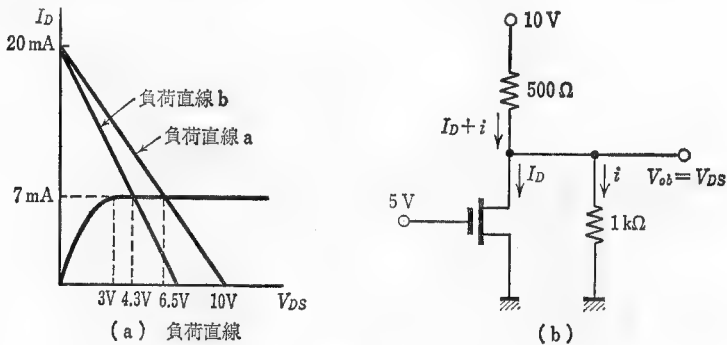



図 P.2

[1.3]　(a)の場合，図P.2(a)の負荷直線aに従ってV_{0a}は

$$V_{0a}=10-500\,\Omega\times7\,\mathrm{mA}=6.5\,\mathrm{V}$$

となる．一方，(b)の場合，

図P.2(b)より

$$V_{DS}=10-500\cdot(I_D+i)=1\,\mathrm{k\Omega}\times i$$

となり，したがって

$$I_D=(20-3\,V_{DS})\times10^{-3}\mathrm{A}$$

I_Dが7mAとなるV_{DS}は4.3Vとなる．すなわち$V_{0b}=4.3\,\mathrm{V}$である．

2 章

[2.1]　加法標準形は

$$f(X,Y,Z)=\bar{X}\bar{Y}\bar{Z}+\bar{X}Y\bar{Z}+\bar{X}YZ$$

乗法標準形は

$$f(X,Y,Z)=(X+Y+\bar{Z})\cdot(\bar{X}+Y+Z)\cdot(\bar{X}+Y+\bar{Z})\cdot(\bar{X}+\bar{Y}+Z)\cdot(\bar{X}+\bar{Y}+\bar{Z})$$

で与えられる．

[2.2]　図P.3のカルノー図より，

$$f(A,B,C)=B\overline{C}+\overline{B}C+AC$$

または，

$$f(A,B,C)=B\overline{C}+\overline{B}C+AB.$$

図 P.3　カルノー図

[2.3]　(1)　表P.1および表P.2より

表 P.1

(a)		(b)
(i)	$\bar{X}\bar{Y}Z$	$\bar{X}Z$……A
	$\bar{X}Y\bar{Z}$	$\bar{Y}Z$……B
(ii)	$\bar{X}YZ$	$\bar{X}Y$……C
	$X\bar{Y}Z$	

表 P.2

	$\bar{X}\bar{Y}Z$	$\bar{X}Y\bar{Z}$	$\bar{X}YZ$	$\bar{X}\bar{Y}Z$	選　択
A	○		○		
B	○			○	◎
C		○	○		◎

$$f(X,Y,Z)=\bar{Y}Z+\bar{X}Y$$

(2)　表P.3およびP.4より

表 P.3

(a)		(b)
(i)	$\bar{X}\bar{Y}\bar{Z}$	$\bar{X}\bar{Y}$……A
(ii)	$\bar{X}\bar{Y}Z$	$\bar{Y}\bar{Z}$……B
	$X\bar{Y}\bar{Z}$	$\bar{Y}Z$……C
(iii)	$X\bar{Y}Z$	$X\bar{Y}$……D
	$XY\bar{Z}$	$X\bar{Z}$……E
		XZ……F
(iv)	XYZ	XY……G

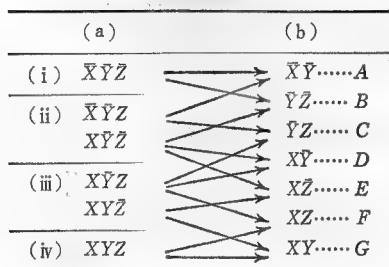



表 P.4

	$\bar{X}\bar{Y}\bar{Z}$	$\bar{X}\bar{Y}Z$	$X\bar{Y}\bar{Z}$	$X\bar{Y}Z$	$XY\bar{Z}$	XYZ	選択
A	○	○					◎
B	○		○				
C		○		○			
D			○	○			◎
E			○		○		
F				○		○	
G					○	○	◎

$$f(X, Y, Z)=A+D+G=\bar{X}\bar{Y}+X\bar{Y}+XY$$
$$=\bar{X}\bar{Y}+X$$

3 章

[3.1] (1) LS-TTL → 標準TTL

(a) 出力Lのとき

$$\text{ファンアウト}=\frac{8\,\text{mA}}{1.6\,\text{mA}}=5$$

(b) 出力Hのとき

$$\text{ファンアウト}=\frac{0.4\,\text{mA}}{40\,\mu\text{A}}=10$$

したがってファンアウトはこれらの小さい方をとって5となる.

(2) HS-CMOS → 標準TTL

(a) 出力Lのとき

$$\text{ファンアウト}=\frac{4\,\text{mA}}{1.6\,\text{mA}}=2.5$$

(b) 出力Hのとき

$$\text{ファンアウト}=\frac{4\,\text{mA}}{40\,\mu\text{A}}=100$$

したがってファンアウトは整数値をとって2.

[3.2] AおよびBがともにHのときのみQ_3はONとなる．同様にCおよびDがともにHのときのみQ_4はONとなる．また，Q_3またはQ_4がONのとき，かつそのときだけ点dはHレベルとなり，Q_6はON，YはLとなる．そのとき，点cはYに対してLレベルとなり，Q_5はOFFとなっている．Q_3およびQ_4がともにOFFのとき点dはL，点cはHとなり，Q_5はON，Q_6はOFFとなる．したがってそのとき，YはHとなる．

これらのことより出力を正論理で考えるとその論理式は

$$Y=\overline{AB+CD} \quad \text{(AND-OR-INVERTER; 図は7451)}$$

で与えられる．

なお，負論理で考えた場合には各々の論理変数に否定をつけ

$$\overline{Y}=\overline{\overline{A}\overline{B}+\overline{C}\overline{D}}$$

より，

$$Y=\overline{A}\overline{B}+\overline{C}\overline{D}$$

となる．

[3.3] (1) 動作可能な電源電圧範囲が広い．

(2) 消費電力が小さい(とくに周波数が高くない範囲で)．

(3) 入力抵抗が高い．

など．

[3.4] 入力が0Vのときを考えると，

$I_E\left(\cong\frac{5}{4+3/\!/1}\right)$はおよそ1mA程度と考えられ，標準TTLを駆動するのに必要な$I_{IL}=1.6$mAに満たない．そのため，出力はLのままとなってしまう．そこで，図P.5のようにすればよい．

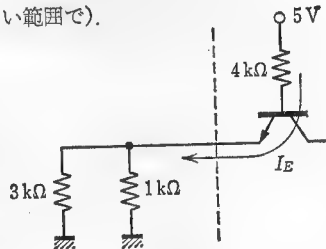



図 P.4

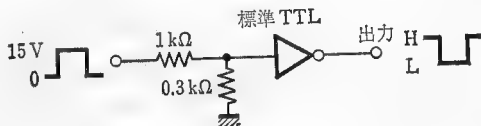



図 P.5

4 章

[4.1] 略

[4.2] 2進4ビット入力ABCDを変換してZ_0~Z_9の10進出力を得るための論理式は

$$\left.\begin{array}{ll} Z_0=\overline{A}\overline{B}\overline{C}\overline{D} & Z_5=\overline{A}B\overline{C}D \\ Z_1=\overline{A}\overline{B}\overline{C}D & Z_6=\overline{A}BC\overline{D} \\ Z_2=\overline{A}\overline{B}C\overline{D} & Z_7=\overline{A}BCD \\ Z_3=\overline{A}\overline{B}CD & Z_8=A\overline{B}\overline{C}\overline{D} \\ Z_4=\overline{A}B\overline{C}\overline{D} & Z_9=A\overline{B}\overline{C}D \end{array}\right\} \qquad \text{(P. 1)}$$

となる．したがって，これから回路を構成すると図P.6(a)のようになる．一方，Z_{10}～Z_{15}は出現しないとしてカルノー図を用いて式(P.1)を簡単化すると

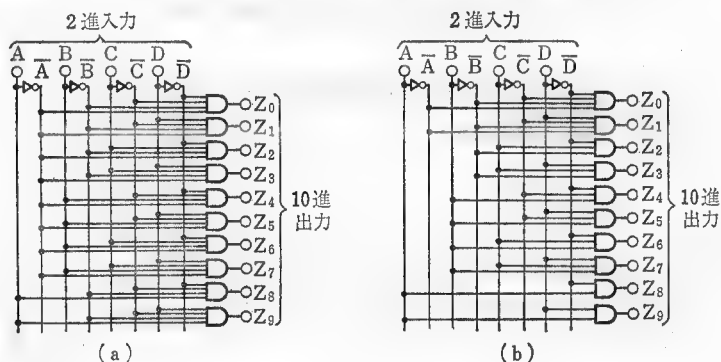



図 P.6

$$\left.\begin{array}{ll} Z_0=\overline{A}\overline{B}\overline{C}\overline{D} & Z_5=B\overline{C}D \\ Z_1=\overline{A}\overline{B}\overline{C}D & Z_6=BC\overline{D} \\ Z_2=\overline{B}C\overline{D} & Z_7=BCD \\ Z_3=\overline{B}CD & Z_8=A\overline{D} \\ Z_4=B\overline{C}\overline{D} & Z_9=AD \end{array}\right\} \qquad \text{(P. 2)}$$

のようになる．これを図示すると，図P.6(b)のようになる．

[4.3]　図P.7参照

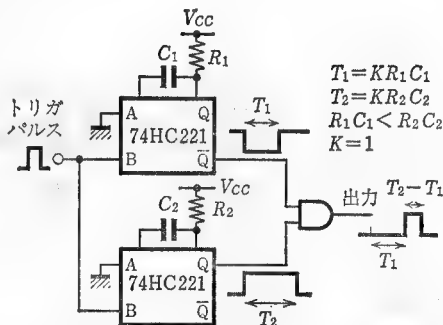



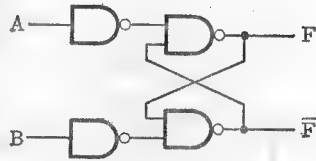
図 P.7

5 章

[5.1] 略

[5.2] A, Bがともに1にならない場合には図P.8のようなRS-FFに変換できるので，状態保持機能を持つ.

図 P.8

[5.3] Gが1のとき，Dは右のRS-FFにつつ抜け状態で伝わり，Qに出力される．Gが0になるとDは伝わらなくなる．したがって，Gが1→0に変化すとるきのDが保存·出力される.

6 章

[6.1] 図P.9, 10参照.

[6.2] 表P.5～7となる．したがって

$$f_1=x+y$$

$$f_0=y+\bar{x}\bar{y}z$$

これらから，図P.11が得られる.

表 P.5　3進プライオリティエンコーダ真理値表

x	y	z	4進プライオリティ出力	2進化プライオリティ出力 f_1	f_0
0	0	0	0	0	0
0	0	1	1	0	1
0	1	0	2	1	0
0	1	1	2	1	0
1	0	0	3	1	1
1	0	1	3	1	1
1	1	0	3	1	1
1	1	1	3	1	1

表 P.6　f_1のカルノー図

xy \ z	0	1
00		
01	1	1
11	1	1
10	1	1

表 P.7　f_0のカルノー図

xy \ z	0	1
00		1
01		
11	1	1
10	1	1

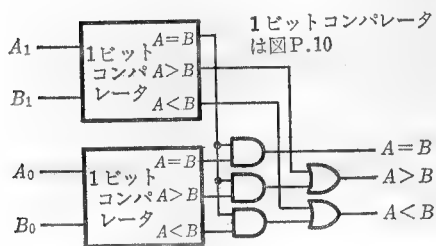
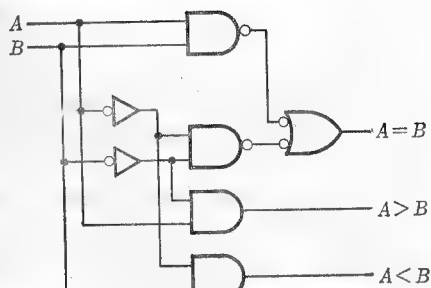



図 P.9　　　　　　図 P.10

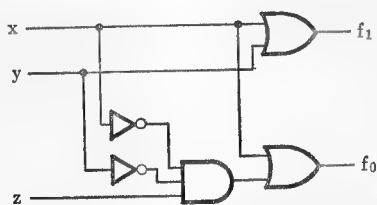



図 P.11

[6.3]　切換えにより，バスの各ラインを入出力の双方向に使えるようにしたもの．使用しないときバスラインから切り離し状態(高インピーダンス状態)にすることができる．

7 章

[7.1]　非同期式カウンタは各段のFFが，たとえば前段の出力の変動によりトリガされるなど，全体として単一のクロックには同期せず動作するカウンタである．一方，同期式カウンタは全体が1つのクロックに同期して動作する．

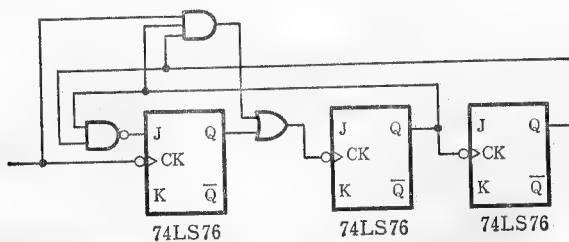
[7.2]　表P.8，図P.12参照

(注)　未使用J, K, $\overline{PR}$, $\overline{CLR}$端子はV_{CC}へ接続しておく．

図 P.12　非同期式7進カウンタ

表 P.8　非同期式7進カウンタの状態遷移

カウント数＼Q_i	2^0 Q_0	2^1 Q_1	2^2 Q_2
6	0	1	1
0	0	0	0
修正措置	(2−a)	(1−b)	(1−a)により不要

表 P.9

カウント	現在の状態			次の状態		
	Q_0	Q_1	Q_2	Q_0'	Q_1'	Q_2'
0	0	0	0	1	0	0
1	1	0	0	0	1	0
2	0	1	0	1	1	0
3	1	1	0	0	0	1
4	0	0	1	0	0	0

表 P.10　Q_0'のカルノー図

Q_0Q_1	Q_2 = 0	Q_2 = 1
0　0	1	0
1　0	0	×
1　1	0	×
0　1	1	×

表 P.11　Q_1'のカルノー図

Q_0Q_1	Q_2 = 0	Q_2 = 1
0　0	0	0
1　0	1	×
1　1	0	×
0　1	1	×

[7.3]　図P.13では，$\overline{\text{CLR}}$端子は使用していない．電源投入時にFFの状態がどうなっ

（注）7474のCLEAR，PRESET端子は，数kΩの抵抗を通して，Vcc（電源）に接続しておく．

図 P.13

表 P. 12　Q_2' のカルノー図

Q_0Q_1	Q_2	
	0	1
0　0	0	0
1　0	0	×
1　1	1	×
0　1	0	×

$Q'_0=\overline{Q}_0\overline{Q}_2$

$Q'_1=Q_0\overline{Q}_1+\overline{Q}_0Q_1$

$Q'_2=Q_0Q_1$

ても，この回路では無限ループに陥ることなく，初期状態(カウント値 0)に移行することが可能である．興味があれば確かめられたい．もちろん，$\overline{\text{CLR}}$端子をGNDに落すクリア回路をつけてもよい．

8 章

[8. 1]　略

[8. 2]　図 P. 14 参照.

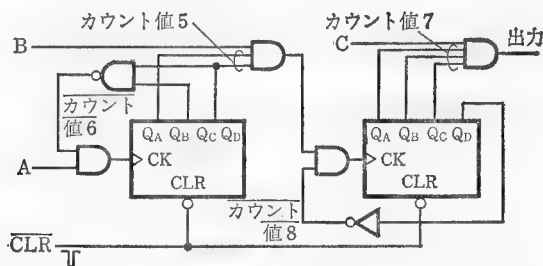



図 P. 14

9 章

[9. 1]　略

[9. 2]　乗算では 0 から

$$(2^8-1)(2^8-1)=2^{16}-2\times2^8+1$$
$$=2^{16}-2^9+1$$

となり，16 ビットを要する.

一方，除算では 0 で割ることを除外すると，正整数の除算 a/b の結果は 0 ～ a となる．したがって答は 8 ビット．

[9. 3]　図 P. 15 参照.

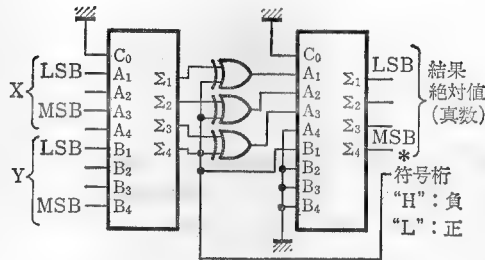
[9. 4]　図 P. 16 参照.

＊オーバフローないとき結果のMSBは１に
なることはない．

図 P. 15

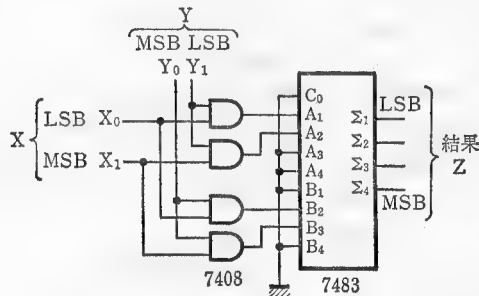



図 P. 16

10　章

[10.1]　略

[10.2]　略

[10.3]　アドレスデータの上位ビットでチップセレクト信号をつくり，メモリチップの選択を行い，下位ビットでチップ内アドレスの指定を行う．

11　章

[11.1]　２重積分形，逐次比較形，並列比較形，各々の説明は略．

[11.2]　標本化定理により，毎秒２W，すなわち40k個の標本化を行う必要がある．これは

$$40\,k[標本/s]\times 12[ビット/標本]=480\,k[ビット/s]$$

のデータとなる．

[11.3]　略．

参 考 文 献

(1) 西野聡，IC論理回路設計の基礎，日刊工業新聞社(昭56)

(2) 田丸啓吉，ハードウェア技術，オーム社(昭59)

(3) 半導体マニュアルシリーズ(年刊)，CQ出版社

最新TTL IC規格表

最新C-MOS IC規格表

最新メモリIC規格表

最新マイコン周辺LSI規格表

最新OPアンプ規格表

最新インターフェース素子規格表

(4) 西野聡，IC論理回路入門，日刊工業新聞社(昭54).

(5) 伊藤誠，基本ハードウェア技術，CQ出版社(昭54).

(6) トランジスタ技術(月刊)，CQ出版社

(7) 電子科学(月刊)，産報出版

(8) インターフェース(月刊)，CQ出版社

(9) 相良岩男，LSIのはなし，日刊工業新聞社(昭57)

(10) 猪瀬博，加藤誠巳，ディジタル回路，産業図書(昭55)

(11) 雨宮好文，ディジタル回路の考えかた，昭晃堂(昭48)

(12) 藤井信生，ディジタル電子回路，昭晃堂(昭62)

(13) 志村正道，電子回路II(ディジタル編)，昭晃堂(昭51)

(14) 松本光功，論理回路，昭晃堂(昭58)

(15) 楠菊信，高木茂，ディジタルシステム，朝倉書店(昭62)

索　引

（五十音順）

マ 行

ラ 行

ワ 行

(ABC 順)

I C 索 引

昭和62年11月27日　初版1刷発行
昭和63年4月30日　初版2刷発行
平成元年1月20日　初版3刷発行
（第1回修正）
平成2年3月20日　初版4刷発行
平成3年3月20日　初版5刷発行
平成3年7月30日　初版6刷発行

著者紹介
田村進一（たむら しんいち）　工学博士
昭和41年　大阪大学基礎工学部電気工学科卒業
昭和46年　大阪大学大学院基礎工学研究科博士課程修了
現　在　大阪大学医学部教授
画像処理，人工知能の研究に従事

検印省略
著者承認

ディジタル回路〔情報工学入門選書4〕
(Digital circuits)

発行者　阿井國昭
東京都新宿区矢来町48
印刷所　創研堂
東京都文京区水道2-7-5
発行所　株式会社　昭晃堂
郵便番号 162　東京都新宿区矢来町48
振替口座　東京3-139320番
電話 (03) 3269-3449(代表)
FAX (03) 3269-1611

定価はカバーに
表示してあります

Printed in Japan
日本書籍出版協会会員
自然科学書協会会員
工学書協会会員

製本　小林共文堂

ISBN4-7856-2024-2